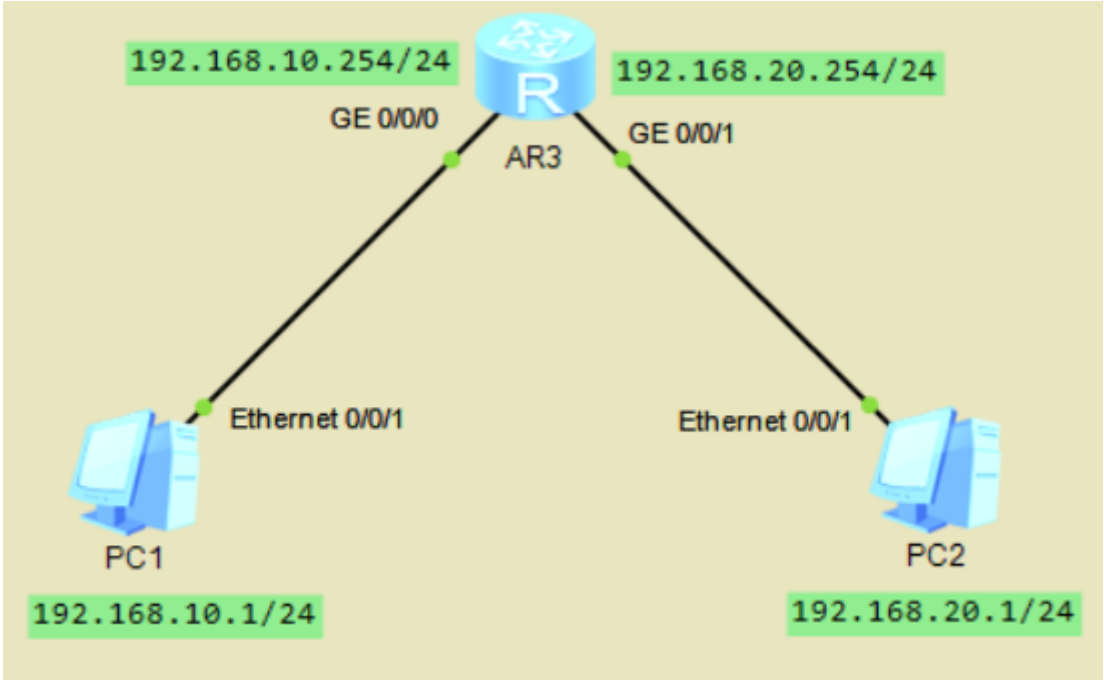
---

```
姓名：坏坏
整理时间：2020年4月27日
```

参考博客：【CSDN】

# ARP代理

**实验**：如下配置两台PC，要求实现两台PC的互相通信。

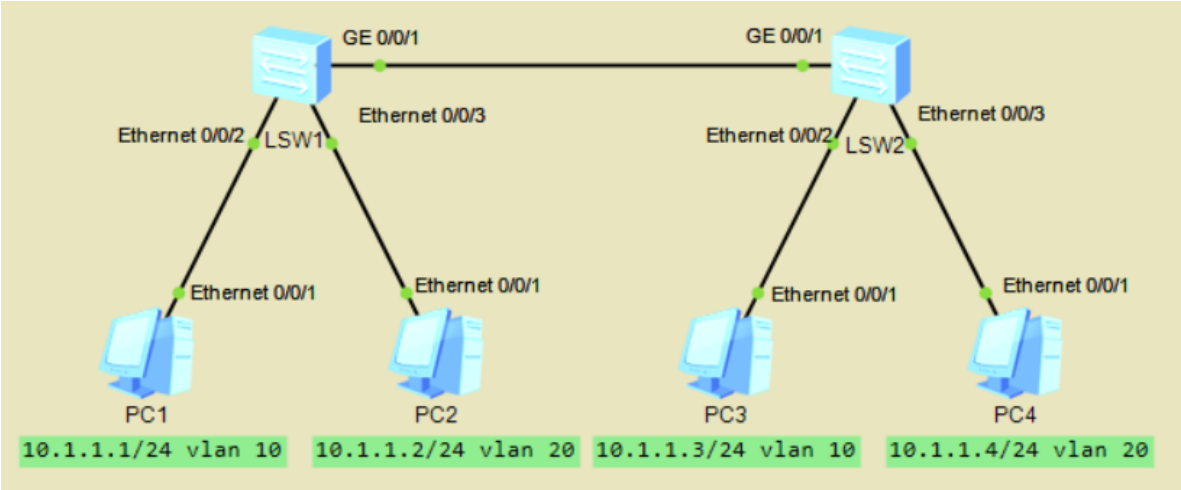



```
为PC各自配置IP，网关设置为G0/0/0口和G0/0/1接口的IP
配置AR3的接口IP，并开启相关的服务

[R1-GigabitEthernet0/0/0]undo info en  //不会提示信息
[R1-GigabitEthernet0/0/0]arp-proxy enable  //开启ARP代理

[R1-GigabitEthernet0/0/1]arp-proxy enable
[R1-GigabitEthernet0/0/1]dis ip int bri  //查看所有的接口信息，检查IP地址是否配上以及接口是否双up
Interface                         IP Address/Mask      Physical   Protocol

GigabitEthernet0/0/0              192.168.10.254/24    up         up
GigabitEthernet0/0/1              192.168.20.254/24    up         up
[R1-GigabitEthernet0/0/1]dis arp all  //查看ARP表项

# 在PC上做连通性测试
```

> 1. 可以通过配置网关实现互通，网关地址为路由器与PC接口的IP
> 2. 通过ARP代理实现互通，需要改变子网掩码使不同网段的IP处于同一网段，如本题中的可以将子网掩码修改为255.255.192.0，即可不通过网关实现互通

# 划分VLAN

**实验**：如下图配置PC的IP地址，需求相同VLAN可以互通，不同VLAN不能互通。

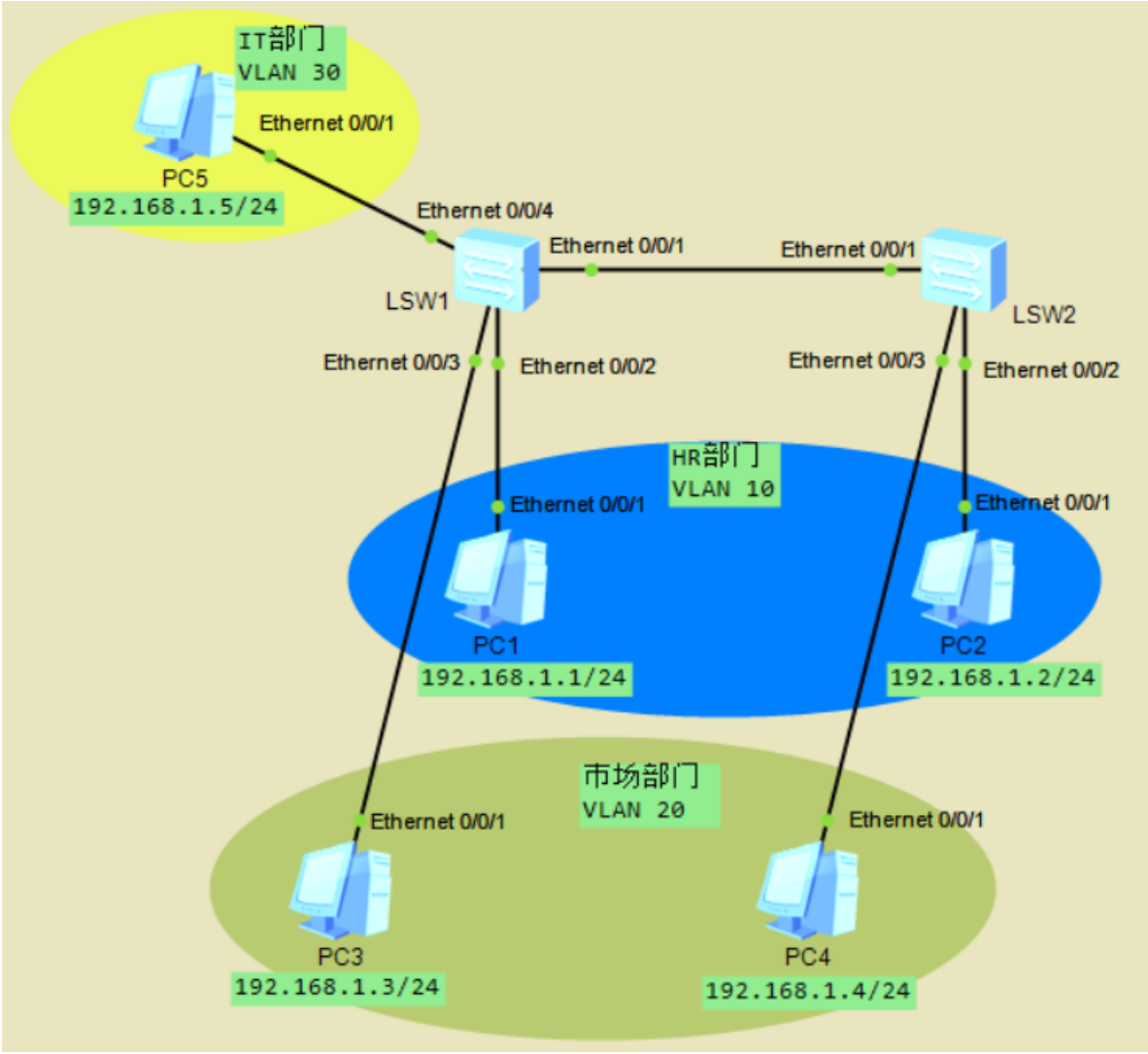



```
[SW1]dis vlan  //查看VLAN
[SW1]vlan batch 10 20  //创建VLAN10、VLAN20
[SW1]int e0/0/2
[SW1-Ethernet0/0/2]port link-type access  //设置接口类型为Access
[SW1-Ethernet0/0/2]port default vlan 10  //默认划分进VLAN10

# 同样方法配置e0/0/3接口，划分进VLAN 20

[SW1]int g0/0/1  //进入g0/0/1接口
[SW1-GigabitEthernet0/0/1]port link-type trunk  //配置接口类型为Trunk
[SW1-GigabitEthernet0/0/1]port trunk allow-pass vlan 10 20  //设置允许通过的VLAN为10 20 ，VLAN1默认允许通过

#SW2相同的配置

#做连通性测试
```

# hybrid

按照如下拓扑，配置相关IP地址。需求：

1. 不同楼层的HR部门和市场部门实现部门内部通信
2. 两部门之间不允许通信
3. IT部门可以访问任意部门



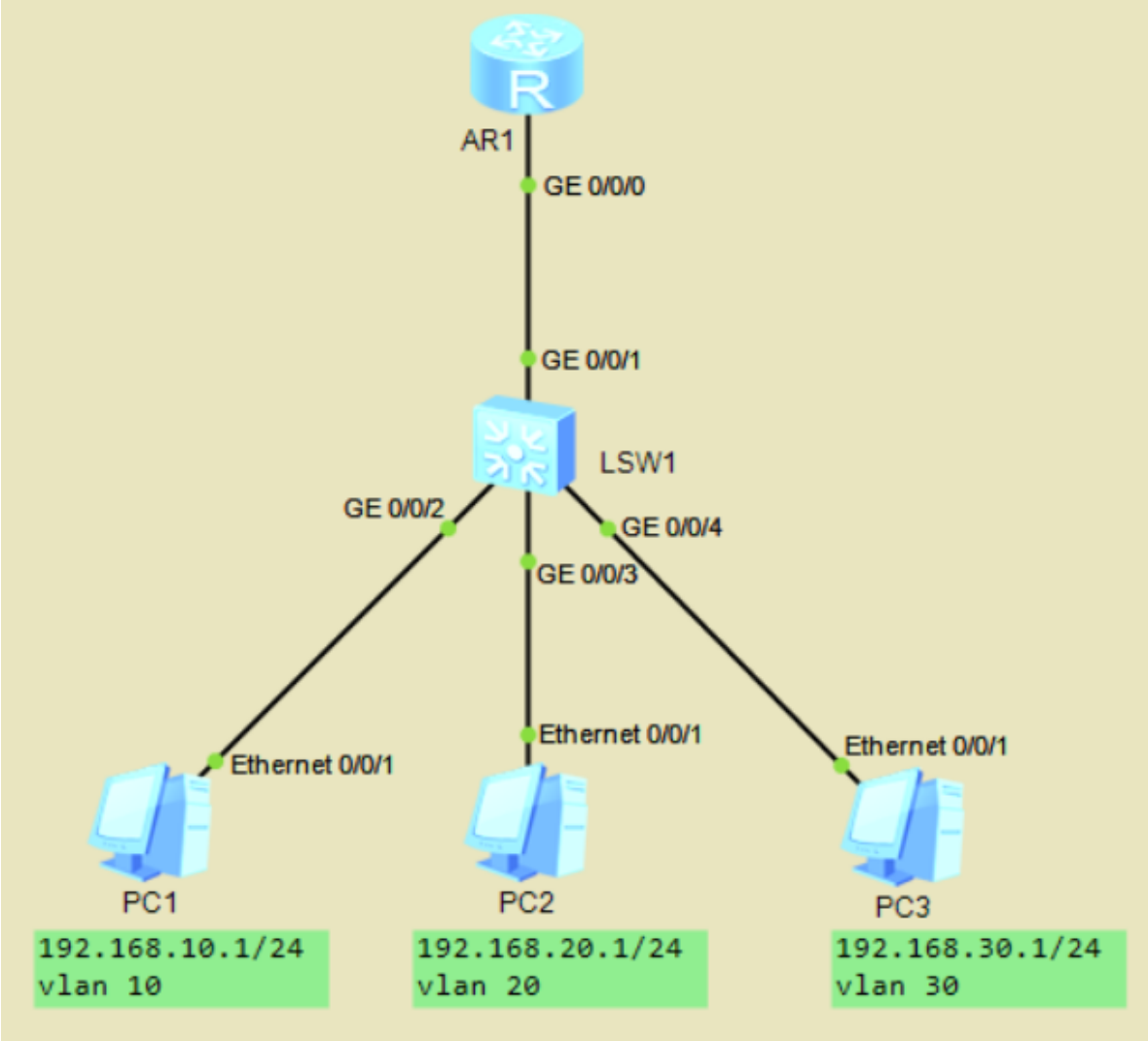

```
[SW1]vlan batch 10 20 30  //创建VLAN10、20、30
[SW1]dis vlan  //查看是否创建
[SW1]int e0/0/3  //进入e0/0/3接口
[SW1-Ethernet0/0/3]port hybrid untagged vlan 20 30  //设置允许通信的VLAN
[SW1-Ethernet0/0/3]port hybrid pvid vlan 20  //设置PVID
[SW1-Ethernet0/0/3]dis th  //查看当前接口下的命令

#同样方法配置e0/0/2接口
port hybrid pvid vlan 10
port hybrid untagged vlan 10 30

#配置e0/0/4接口
port hybrid pvid vlan 30
port hybrid untagged vlan 10 20 30

#配置e/0/1接口
port hybrid tagged vlan 10 20 30
默认PVID是VLAN 1

#Sw2同样的配置

#进行连通性测试
```

# VLAN间路由

# 单臂路由

1. 把PC划分到相应的VLAN
2. 把g0/0/1接口配置成trunk，并允许所有VLAN通过
3. 配置路由器的子接口配置IP地址
4. 子接口配置VLAN ID封装（`dot1q termination vid 10`）
5. 接口开启arp广播（`arp broadcast enable`）

如下拓扑图，为PC配置IP地址。配置单臂路由，实现PC间互通。

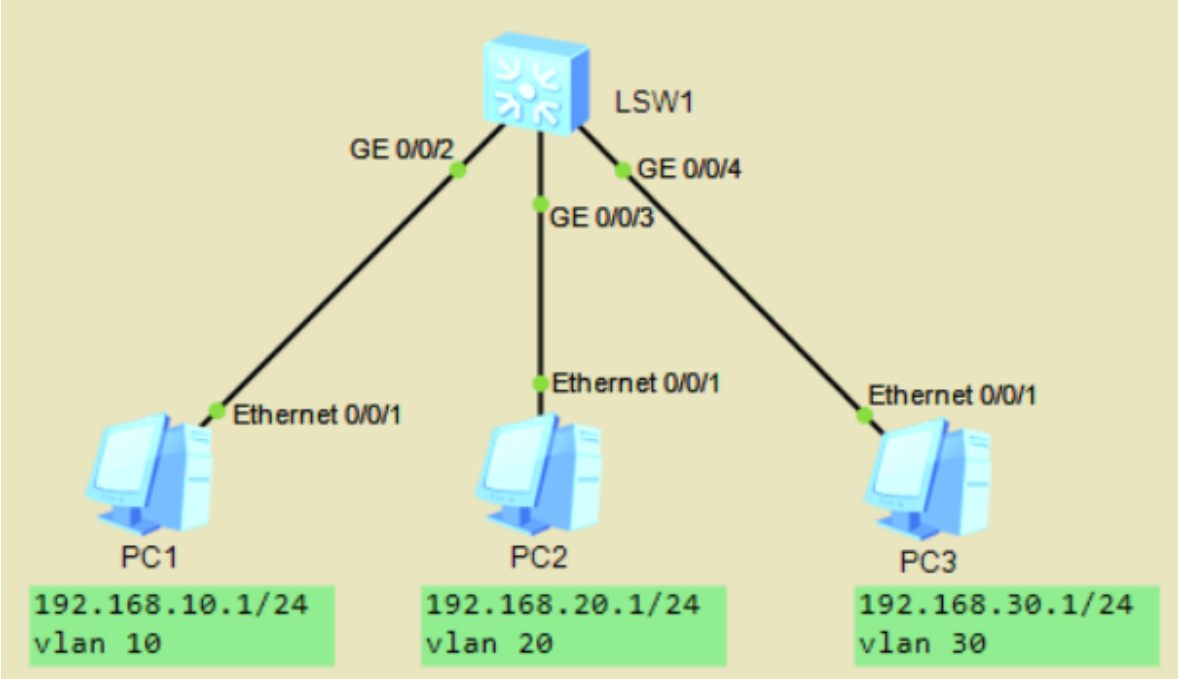



```
#先给PC配置相应的IP地址，网关254

[SW1]vlan batch 10 20 30  //创建VLAN
[SW1]dis vlan  //查看VLAN是否创建成功
[SW1]int g0/0/2  //进入g0/0/2接口
[SW1-GigabitEthernet0/0/2]port link-type access  //配置接口类型为access
[SW1-GigabitEthernet0/0/2]port default vlan 10  //划分默认VLAN

#同样的方法配置g0/0/3、g0/0/4接口

#配置trunk接口，并允许所有VLAN通过
[SW1]int g0/0/1
[SW1-GigabitEthernet0/0/1]port link-type trunk  //配置接口类型为trunk
[SW1-GigabitEthernet0/0/1]port trunk all vlan all  //允许所有VLAN通过

#在R1上配置子接口
```

```
[R1]int g0/0/0.1  //配置子接口
[R1-GigabitEthernet0/0/0.1]ip add 192.168.10.254 24  //为子接口配置IP
#同样方法配置其他子接口
[R1]dis ip int br  //查看所有接口详细信息

#封装VLAN号
[R1]int g0/0/0.1
[R1-GigabitEthernet0/0/0.1]dot1q termination vid 10  //指定vid，即这个接口对应的VLAN ID
[R1-GigabitEthernet0/0/0.1]arp broadcast enable  //开启ARP的广播功能

#同样方法配置其他的子接口

#进行连通性测试
```

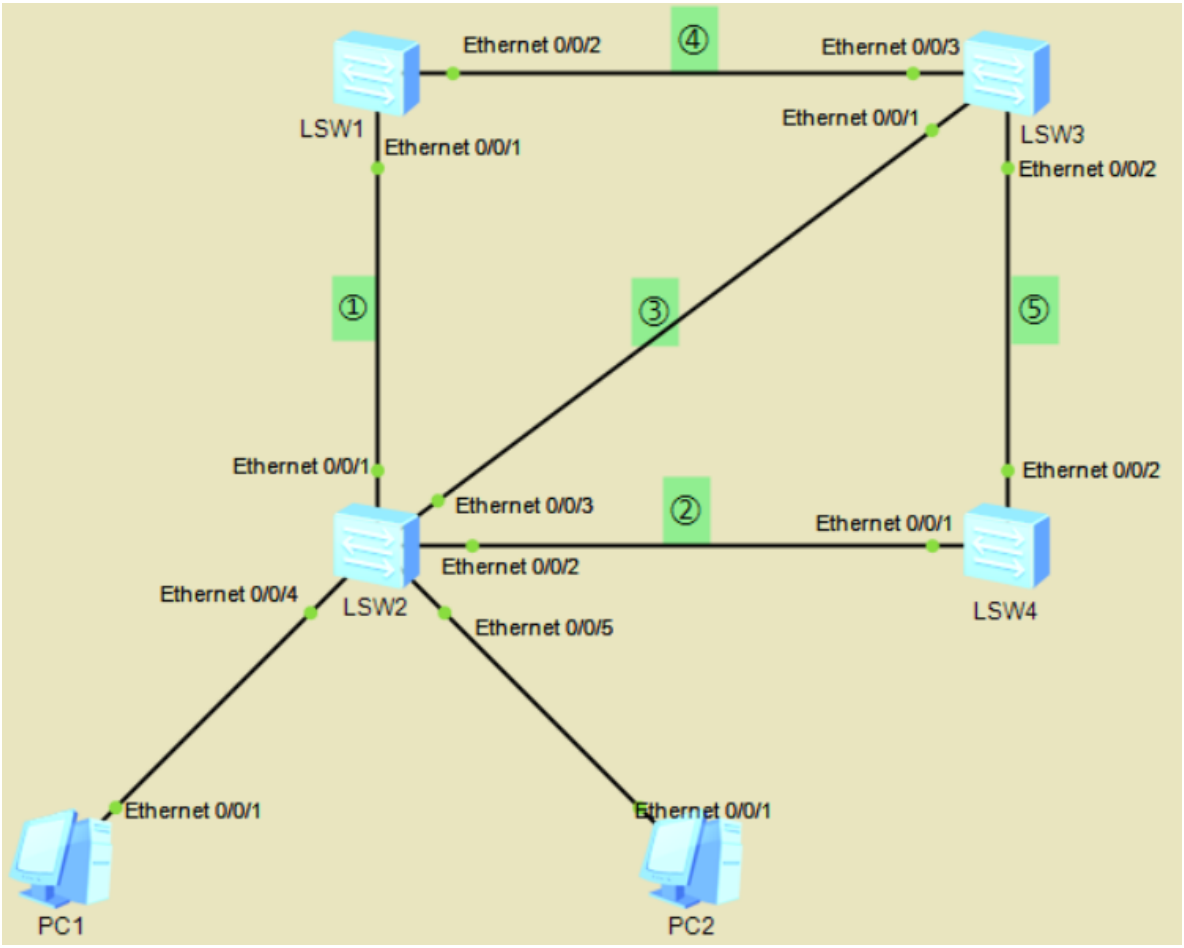
# 三层交换

**实验**：如下拓扑图，配置相应IP地址。配置三层交换，使PC间互通。




```
[SW1]int Vlanif 10  //创建VLAN10
[SW1-Vlanif10]ip add 192.168.10.254 24  //配置IP地址
[SW1-Vlanif10]int vlanif 20
[SW1-Vlanif20]ip add 192.168.20.254 24
[SW1-Vlanif20]int vlanif 30
[SW1-Vlanif30]ip add 192.168.30.254 24

#进行连通性测试
```

# STP配置




SW1：4c1f-cc5c-74c7
SW2：4c1f-cc2d-7013
SW3：4c1f-cc80-7370
SW4：4c1f-cc6f-1691

1. 选举根桥
    - 交换BPDU，比较BPDU，相同
    - 比较MAC地址，SW2的MAC最小，选举为根桥
2. 选举根端口
    - 比较路径开销，SW1在1号线路到达根桥路径开销最小，所以SW1的1接口为RP（同理SW3的1接口、SW4的1接口都为RP）
    - 如果路径开销相同，比较BID（优先级、MAC地址）
    - 如果BID也相同，则比较PID（优先级、端口号）
3. 选举指定端口
    - 在网络上（每条线路上）选举指定端口
    - 根桥开销为0，所以SW2的1、2、3接口都为DP
    - 4号线路上走1、3线路开销相同，比较BID（优先级、MAC地址），SW1的MAC地址小，则SW1的2接口为DP，SW3的3接口为AP
    - 5号线路上走2、3线路开销相同，比较BID（优先级、MAC地址），SW4的MAC地址小，则SW4的2接口为DP，SW3的2接口为AP

```
# 查看MAC地址
[SW1]dis stp  //查看MAC地址

[SW1]dis stp bri  //查看SW1的STP
 MSTID  Port                        Role  STP State     Protection
   0    Ethernet0/0/1               ROOT  FORWARDING      NONE
   0    Ethernet0/0/2               DESI  FORWARDING      NONE
```

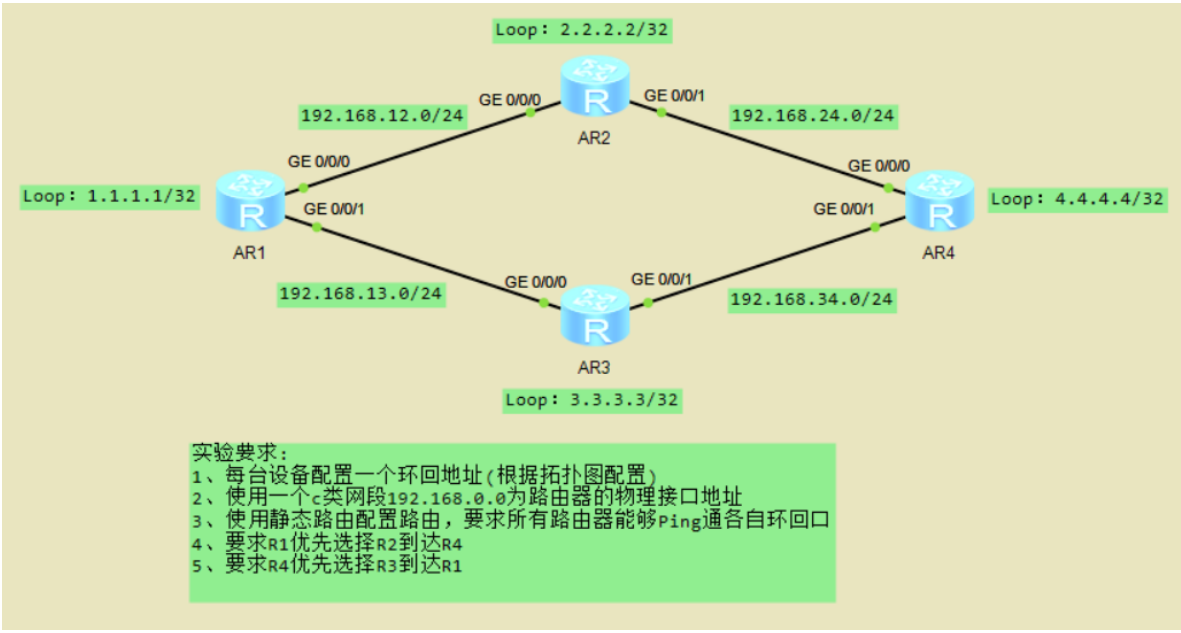
```
# Ethernet0/0/1为RP，FORWARDING为正常转发数据，Ethernet0/0/2为DP

[SW2]dis stp bri  //查看SW2的STP
 MSTID  Port                        Role  STP State     Protection
   0    Ethernet0/0/1               DESI  FORWARDING      NONE
   0    Ethernet0/0/2               DESI  FORWARDING      NONE
   0    Ethernet0/0/3               DESI  FORWARDING      NONE
   0    Ethernet0/0/4               DESI  FORWARDING      NONE
   0    Ethernet0/0/5               DESI  FORWARDING      NONE

[SW3]dis stp bri  //查看SW3的STP
 MSTID  Port                        Role  STP State     Protection
   0    Ethernet0/0/1               ROOT  FORWARDING      NONE
   0    Ethernet0/0/2               ALTE  DISCARDING      NONE
   0    Ethernet0/0/3               ALTE  DISCARDING      NONE
# Ethernet0/0/1为RP，数据正常转发，Ethernet0/0/2和Ethernet0/0/3为AP，DISCARDING端口关闭，不转发数据

[SW4]dis stp bri  //查看SW4的STP
 MSTID  Port                        Role  STP State     Protection
   0    Ethernet0/0/1               ROOT  FORWARDING      NONE
   0    Ethernet0/0/2               DESI  FORWARDING      NONE
```

**拓展**：使SW1为根桥，SW3位次根桥

```
[SW1]stp root primary  //使SW1成为主根桥
[SW1]dis stp  //查看cost优先级为0


[SW3]stp root secondary  //使SW3成为次根桥
[SW3]dis stp  //查看cost优先级为4096

# 增长为12次方增长，下一个是8192，一次类推

[SW1]int e0/0/1
[SW1-Ethernet0/0/1]stp cost ?  //修改接口开销
  INTEGER<1-200000000>  Port path cost
[SW1-Ethernet0/0/1]stp cost 55
```

# 静态路由协议

**实验**：如下拓扑，按照图上要求配置IP。



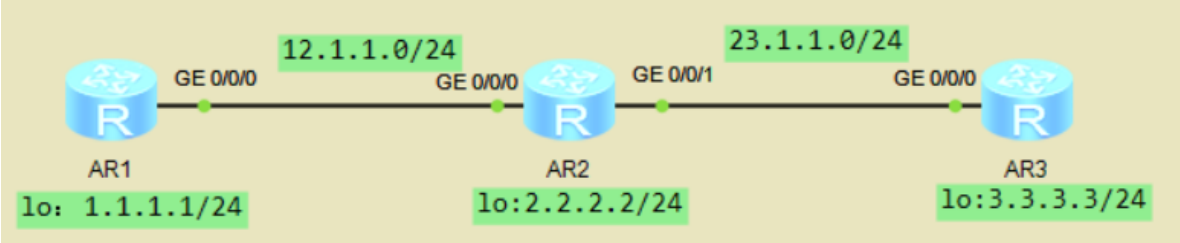

```
# 配置本地环回口地址
[R1]int LoopBack 1
[R1-LoopBack1]ip ad 4.4.4.4 32

# R1上的静态路由配置
ip route-static 2.2.2.2 255.255.255.255 192.168.12.2
ip route-static 3.3.3.3 255.255.255.255 192.168.13.3
ip route-static 4.4.4.4 255.255.255.255 192.168.12.2 preference 10
ip route-static 4.4.4.4 255.255.255.255 192.168.13.3 preference 100
ip route-static 192.168.24.0 255.255.255.0 192.168.12.2
ip route-static 192.168.34.0 255.255.255.0 192.168.12.2

# R2上的静态路由配置
ip route-static 1.1.1.1 255.255.255.255 192.168.12.1
ip route-static 3.3.3.3 255.255.255.255 192.168.12.1
ip route-static 4.4.4.4 255.255.255.255 192.168.24.4
ip route-static 192.168.13.0 255.255.255.0 192.168.12.1
ip route-static 192.168.34.0 255.255.255.0 192.168.24.4

# R3上的静态路由配置
ip route-static 1.1.1.1 255.255.255.255 192.168.13.1
ip route-static 2.2.2.2 255.255.255.255 192.168.13.1
ip route-static 4.4.4.4 255.255.255.255 192.168.34.4
ip route-static 192.168.12.0 255.255.255.0 192.168.13.1
ip route-static 192.168.24.0 255.255.255.0 192.168.34.4

# R4上的静态路由配置
ip route-static 1.1.1.1 255.255.255.255 192.168.34.3 preference 10
ip route-static 2.2.2.2 255.255.255.255 192.168.24.2
ip route-static 3.3.3.3 255.255.255.255 192.168.34.3
ip route-static 192.168.12.0 255.255.255.0 192.168.34.3 preference 10
ip route-static 192.168.12.0 255.255.255.0 192.168.24.2
ip route-static 192.168.13.0 255.255.255.0 192.168.34.3 preference 10

#进行连通性测试，Tracer跟踪查看数据转发路径
```

> - `save`保存配置，重启后配置依旧生效
> - 用户视图下执行`reset saved-configuration`（清空所有配置），然后`reboot`重启

# 动态路由协议

## RIP配置

**实验**：如图配置IP地址




```
# 配置环回接口地址与物理接口地址
[R1]int LoopBack  1
[R1-LoopBack1]ip ad 1.1.1.1 24
[R1-LoopBack1]int g0/0/0
[R1-GigabitEthernet0/0/0]ip ad 12.1.1.1 24
# 相同方法配置其他路由器

# 配置RIP，对外宣告主网（宣告的为自身已知的主网）
[R1]rip 1
[R1-rip-1]network 1.0.0.0
[R1-rip-1]network 12.0.0.0
#相同方法配置其他路由器

# 连通性测试
```

```
# 配置RIP认证方式
[R1]int g0/0/0
[R1-GigabitEthernet0/0/0]rip authentication-mode simple cipher huawei
[R1-GigabitEthernet0/0/0]q
[R1]
```

**RIP环路**

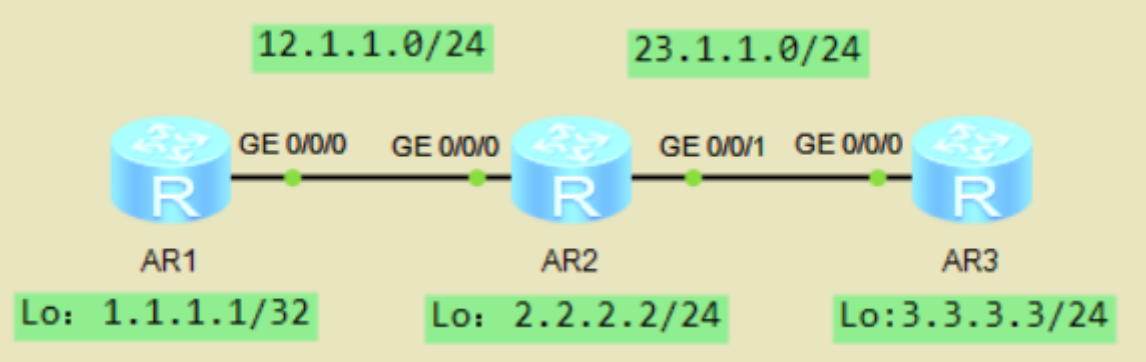
- 网络发生故障时，RIP网络有可能会产生环路
- 环路避免：
  - **水平分割**：路由器从某个接口学到的路由，不会从该接口再发回给领居路由
  - **毒性逆转**：路由从某个接口学到路由后，将该路由的跳数设置为16，并从原接收接口发回给领居路由器
  - **触发更新**：当路由信息发生变化时，立即向邻居设备发送触发更新报文（避免环路产生）

```
# RIP配置
[R1]rip  //进入RIP协议视图
[R1-rip-1]version 2  //更改v2的版本
[R1-rip-1]network 10.0.0.0  //对外宣告主网

# 配置Metricin（度量值）
```

```
[R1]int g0/0/0
[R1-GigabitEthernet0/0/0]rip metricin 2  //更改进接口的度量值
[R1-GigabitEthernet0/0/0]rip metricout 2  //更改出接口的度量值

# 水平分割 & 毒性逆转
[R1-GigabitEthernet0/0/0]rip split-horizon  //配置水平分割，默认开启
[R1-GigabitEthernet0/0/0]rip poison-reverse  //配置毒性逆转，默认开启
# 当两个特性都配置时，只有毒性逆转会生效

# 配置RIP报文的收发
[R1-GigabitEthernet0/0/0]undo rip output  //禁止发送RIP报文
[R1-GigabitEthernet0/0/0]undo rip input  //禁止接收RIP报文

# 抑制接口，命令优先级大于rip in/output
[R1]rip  //进入接口视图
[R1-rip-1]silent-interface g0/0/0  //抑制接口，只接受RIP报文，不发送
```

# OSPF

**实验一**：如图配置IP，配置OSPF，要求R1、R2、R3互通。



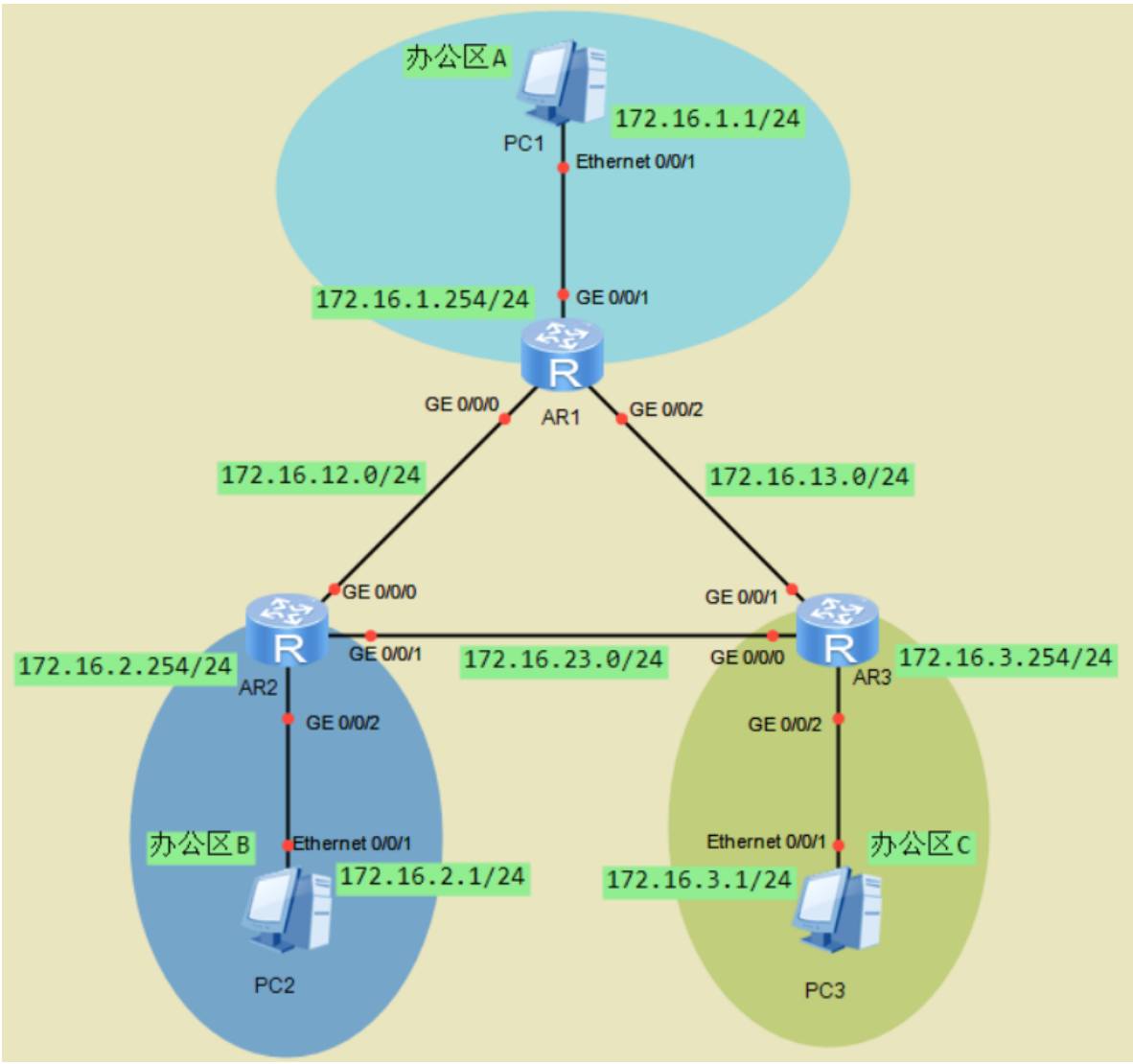

```
# R1
[R1]ospf 1  //指定OSPF的进程号1
[R1-ospf-1]area 0  //进入骨干区域
[R1-ospf-1-area-0.0.0.0]network 12.1.1.0 0.0.0.255  //宣告网段
[R1-ospf-1-area-0.0.0.0]net 1.1.1.1 0.0.0.0  //宣告精确地址
# R2
[R2]ospf 1
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]net 2.2.2.2 0.0.0.0
[R2-ospf-1-area-0.0.0.0]net 12.1.1.2 0.0.0.0
[R2-ospf-1-area-0.0.0.0]net 23.1.1.2
[R2-ospf-1-area-0.0.0.0]net 23.1.1.2 0.0.0.0

# 查看邻居关系
[R1]dis ospf peer bri  //查看邻居关系

# R3
[R3]ospf
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]net 3.3.3.3 0.0.0.0
[R3-ospf-1-area-0.0.0.0]net 23.1.1.3 0.0.0.0
# 连通性测试
```

- 指定Router-id

```
# 如果没有手动指定router-id会自动选取
[R2]router id 12.1.1.2  //手动指定Router-ID
<R2>reset ospf process  //重新启动OSPF进程
[R2]dis ospf peer bri  //查看邻居关系，Router-ID变成了指定的12.1.1.2
[R2]dis ospf int g0/0/0  //查看接口下的OSPF
```

# OSPF单区域

如图配置IP地址，需求使用OSPF配置，实现全网互通。



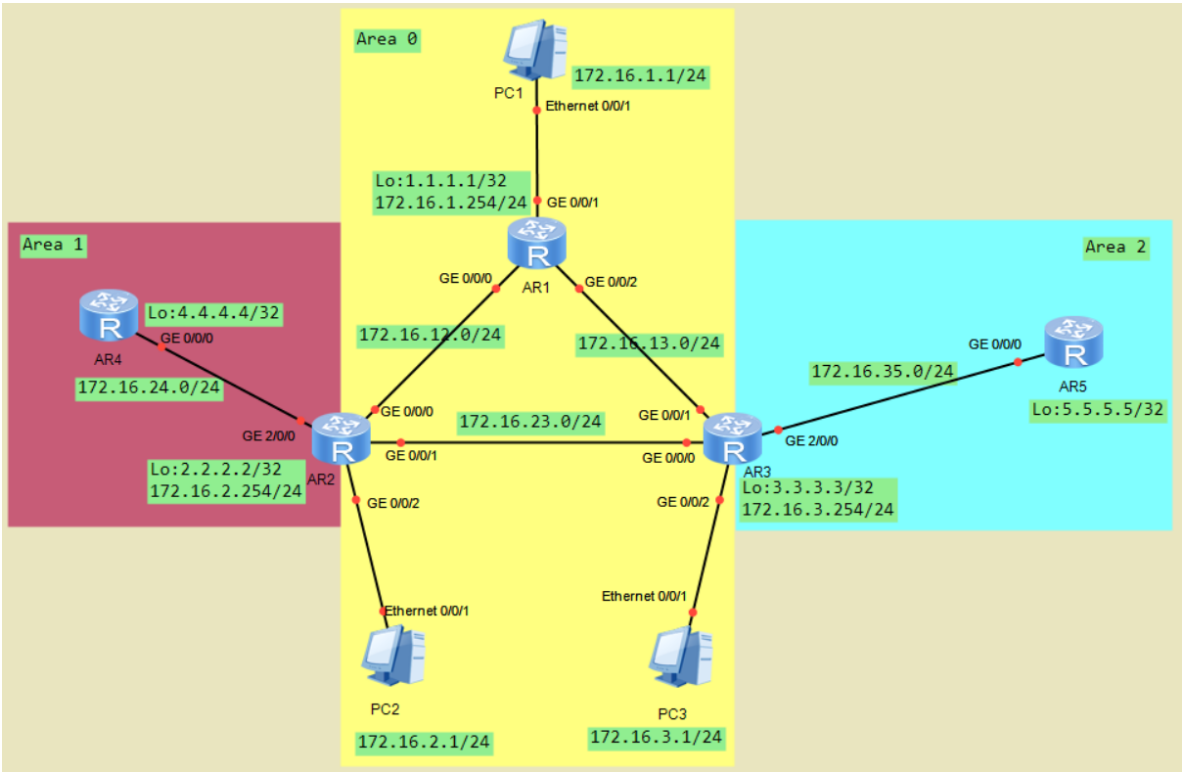

- 配置OSPF

```
# R1
[R1]ospf router-id 1.1.1.1  //手动指定Router-id
[R1-ospf-1]area 0  //进入骨干区域
[R1-ospf-1-area-0.0.0.0]net 1.1.1.1 0.0.0.0  //精确宣告1.1.1.1
[R1-ospf-1-area-0.0.0.0]net 172.16.1.254 0.0.0.0  //精确宣告172.16.1.254
[R1-ospf-1-area-0.0.0.0]net 172.16.13.1 0.0.0.0  //精确宣告172.16.13.1
[R1-ospf-1-area-0.0.0.0]net 172.16.12.1 0.0.0.0  //精确宣告172.16.12.1

# R2
[R2]ospf router-id 2.2.2.2
[R2-ospf-1]area 0
```

```
[R2-ospf-1-area-0.0.0.0]net 2.2.2.2 0.0.0.0
[R2-ospf-1-area-0.0.0.0]net 172.16.2.254 0.0.0.0
[R2-ospf-1-area-0.0.0.0]net 172.16.23.2 0.0.0.0
[R2-ospf-1-area-0.0.0.0]net 172.16.12.2 0.0.0.0

# R3
[R3]ospf router-id 3.3.3.3
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]net 3.3.3.3 0.0.0.0
[R3-ospf-1-area-0.0.0.0]net 172.16.3.254 0.0.0.0
[R3-ospf-1-area-0.0.0.0]net 172.16.23.3 0.0.0.0
[R3-ospf-1-area-0.0.0.0]net 172.16.13.3 0.0.0.0

[R1]dis cu conf ospf  //查看OSPF的所有配置
[R1]dis ospf peer bri  //查看OSPF的邻居状态
[R1]dis ip routing-table protocol ospf  //查看OSPF学习到的路由表

# 连通性测试
```

# OSPF多区域

OSPF多区域配置，需求全网互通




- 配置OSPF

```
# R1
[R1]ospf router-id 1.1.1.1  //手动指定Router-id
[R1-ospf-1]area 0  //进入骨干区域
[R1-ospf-1-area-0.0.0.0]net 1.1.1.1 0.0.0.0  //精确宣告1.1.1.1
[R1-ospf-1-area-0.0.0.0]net 172.16.1.254 0.0.0.0  //精确宣告172.16.1.254
[R1-ospf-1-area-0.0.0.0]net 172.16.13.1 0.0.0.0  //精确宣告172.16.13.1
[R1-ospf-1-area-0.0.0.0]net 172.16.12.1 0.0.0.0  //精确宣告172.16.12.1

```

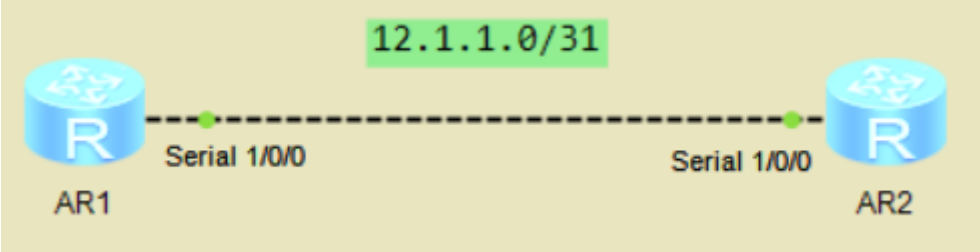
```
# R2
[R2]ospf router-id 2.2.2.2
[R2-ospf-1]area 0
[R2-ospf-1-area-0.0.0.0]net 2.2.2.2 0.0.0.0
[R2-ospf-1-area-0.0.0.0]net 172.16.2.254 0.0.0.0
[R2-ospf-1-area-0.0.0.0]net 172.16.23.2 0.0.0.0
[R2-ospf-1-area-0.0.0.0]net 172.16.12.2 0.0.0.0
[R2-ospf-1-area-0.0.0.0]q
[R2-ospf-1]area 1
[R2-ospf-1-area-0.0.0.1]net 172.16.24.2 0.0.0.0

# R3
[R3]ospf router-id 3.3.3.3
[R3-ospf-1]area 0
[R3-ospf-1-area-0.0.0.0]net 3.3.3.3 0.0.0.0
[R3-ospf-1-area-0.0.0.0]net 172.16.3.254 0.0.0.0
[R3-ospf-1-area-0.0.0.0]net 172.16.23.3 0.0.0.0
[R3-ospf-1-area-0.0.0.0]net 172.16.13.3 0.0.0.0
[R3-ospf-1-area-0.0.0.0]q
[R3-ospf-1]area 2
[R3-ospf-1-area-0.0.0.2]net 172.16.35.3 0.0.0.0

# R4
[R4]ospf 1  //进入OSPF进程
[R4-ospf-1]area 1  //进入区域1
[R4-ospf-1-area-0.0.0.1]net 4.4.4.4 0.0.0.0  //精确宣告IP地址
[R4-ospf-1-area-0.0.0.1]net 172.16.24.4 0.0.0.0

# R5
[R5]ospf 1
[R5-ospf-1]area 2  //进入区域2
[R5-ospf-1-area-0.0.0.2]net 5.5.5.5 0.0.0.0  //精确宣告
[R5-ospf-1-area-0.0.0.2]net 172.16.35.5 0.0.0.0

# 显示当前学习到的LSA信息
[R2]dis ospf lsdb  //查看连接的数据库

# 连通性测试
```

## OSPF开销&认证

```
# 修改cost值
[R1]interface GigabitEthernet 0/0/0
[R1-GigabitEthernet0/0/0]ospf cost 20

# 修改带宽
[R1]ospf
[R1-ospf-1]bandwidth-reference 10000

# 基于接口认证
[R1]interface GigabitEthernet0/0/0
[R1-GigabitEthernet0/0/0]ospf authentication-mode md5 1 cipher huawei
```

# HDLC配置

如图配置IP地址，使用HDLC接口调用配置接口。

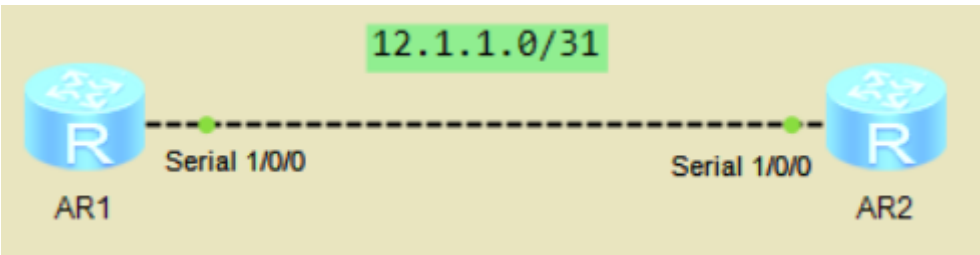



```
# R1修改端口协议
[R1]int s1/0/0
[R1-Serial1/0/0]link-protocol hdlc   //修改为hdlc协议

# R2修改端口协议
[R2]int s1/0/0
[R2-Serial1/0/0]link-protocol hdlc

# 配置环回口地址
[R1]int lo 1
[R1-LoopBack1]ip ad 12.1.1.1 32  //配置环回口地址

[R1]int s1/0/0
[R1-Serial1/0/0]ip address unnumbered interface LoopBack 1  //接口借用

# 添加静态路由
[R1]ip route-static 12.1.1.0 30 s1/0/0

# 连通性测试
```

# PPP

## PAP认证

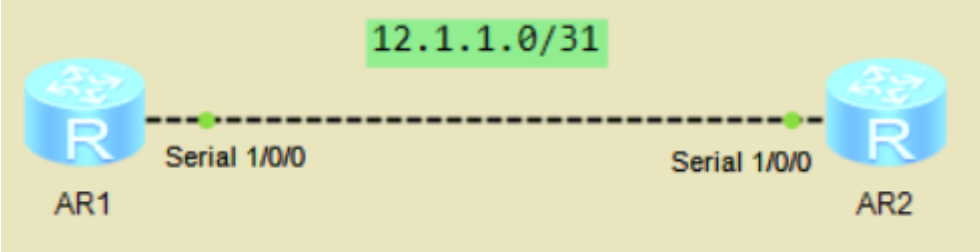
如图拓扑，配置IP地址，配置PPP的PAP认证。



```
[R1]int s1/0/0
[R1-Serial1/0/0]ppp authentication-mode pap  //PPP认证模式修改为PAP

# AAA认证
[R1]aaa
[R1-aaa]local-user bad password cipher huawei123  //配置用户名和密码
[R1-aaa]local-user bad service-type ppp  //配置用户用于PPP

# R2上配置
[R2]int s1/0/0
[R2-Serial1/0/0]ppp pap local-user bad password cipher huawei123  //被认证方认证
# 连通性测试
```

# Chap认证

如图配合IP地址，配置PPP的Chap认证。



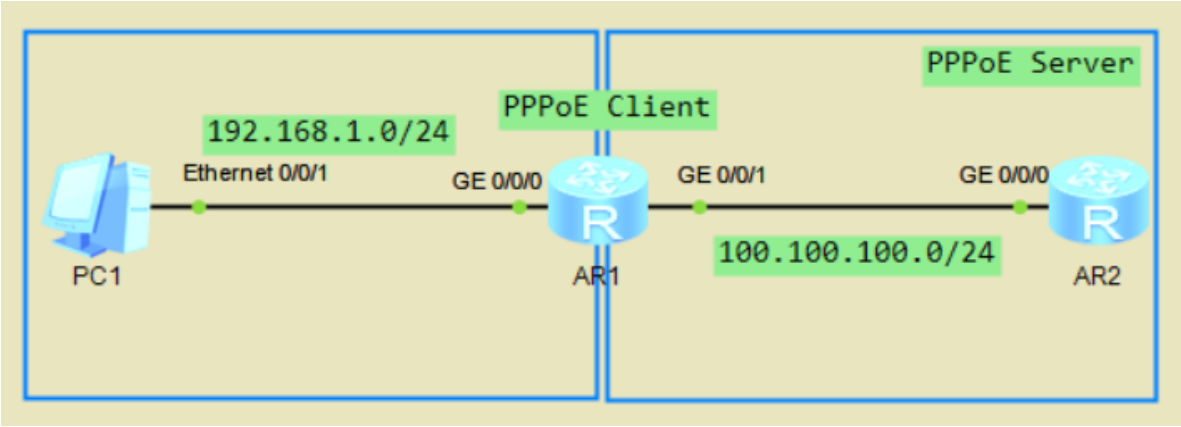

- R1、R2配置接口IP地址

```
# 配置Chap认证
[R1]int s1/0/0
[R1-Serial1/0/0]ppp authentication-mode chap

# 配置AAA认证
[R1]aaa
[R1-aaa]local-user bad password cipher huawei123  //配置用户名和密码
[R1-aaa]local-user bad service-type ppp  //配置用户用于PPP

#
[R2]int s1/0/0
[R2-Serial1/0/0]ppp chap user bad
[R2-Serial1/0/0]ppp chap password cipher huawei123

# 连通性测试
```

# PPPoE配置

- **PPPoE Server配置步骤**
  - 创建Dialer接口并通过配置IP地址
  - 配置PAP认证
  - 绑定拨号接口
  - 查看被分配的IP地址

如下拓扑，配置PPPoE，使PC与PPPoE Server互通




- PPPoE Server配置

```
# 创建并配置虚拟模板
[PPPoE Server]int Virtual-Template 1  //创建虚拟模板
[PPPoE Server-Virtual-Template1]ip ad 100.100.100.254 24  //虚拟模板配置IP地址
[PPPoE Server-Virtual-Template1]ppp ipcp dns 8.8.8.8  //配置DNS

# 创建并配置地址池
[PPPoE Server]ip pool pppoe  //创建地址池
[PPPoE Server-ip-pool-pppoe]network 100.100.100.0 mask 24  //分配网段
[PPPoE Server-ip-pool-pppoe]gateway-list 100.100.100.254  //设置网关

# 虚拟模板调用地址池并配置认证
[PPPoE Server]int Virtual-Template 1  //进入虚拟模板接口
[PPPoE Server-Virtual-Template1]remote address pool pppoe  //调用地址池
[PPPoE Server-Virtual-Template1]ppp authentication-mode pap  //配置认证模式

# 物理接口绑定虚拟模板接口
[PPPoE Server]int g0/0/0
[PPPoE Server-GigabitEthernet0/0/0]pppoe-server bind virtual-template 1  //
物理接口绑定虚拟模板

# 配置AAA认证
[PPPoE Server]aaa
[PPPoE Server-aaa]local-user bad password cipher huawei123
[PPPoE Server-aaa]local-user bad service-type ppp
```

- PPPoE Client配置

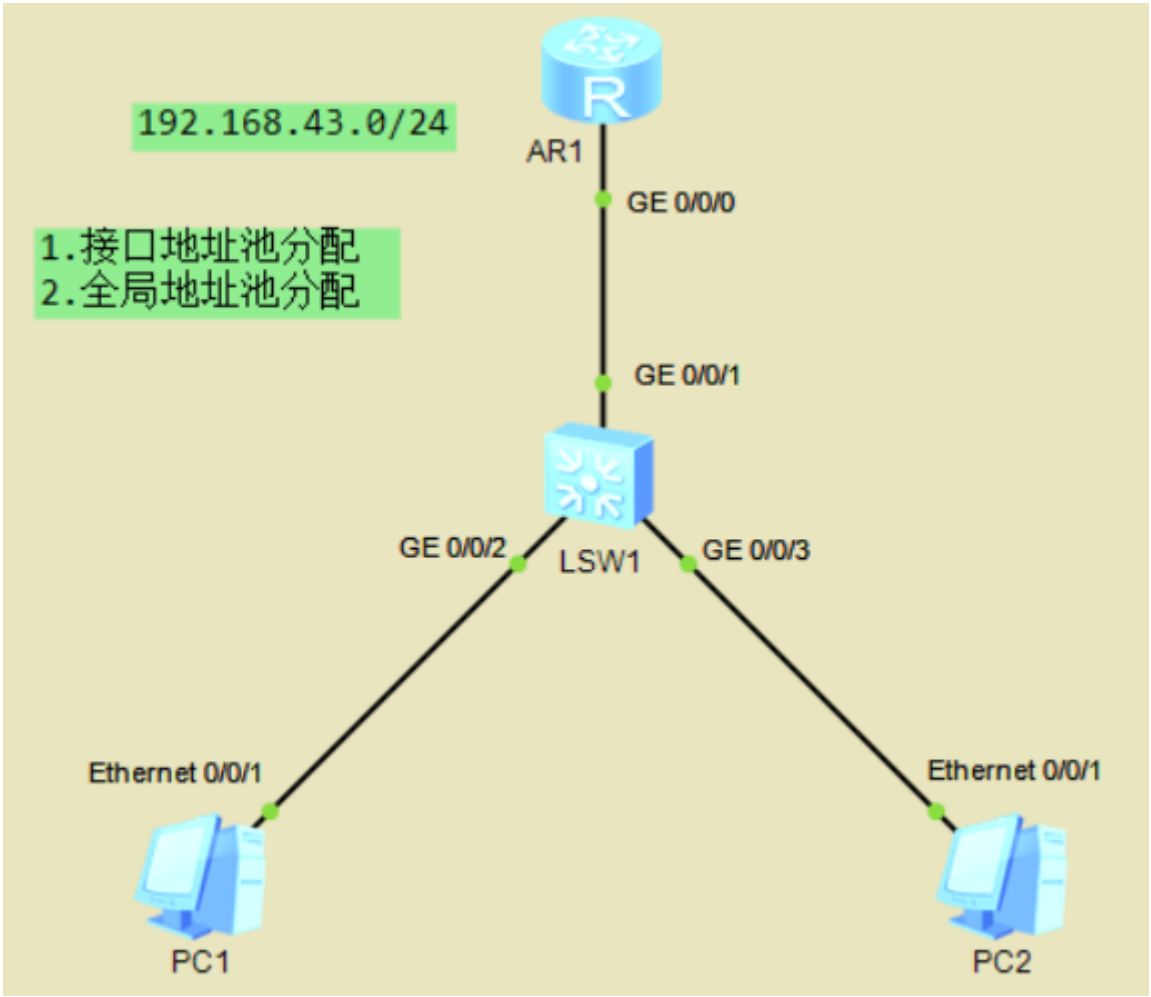
```
# 创建Dialer接口并通过配置IP地址
[PPPoE Client]int Dialer 1  //创建Dialer接口
[PPPoE Client-Dialer1]dialer user bad  //指定Dialer用户（可配可不配）
[PPPoE Client-Dialer1]dialer bundle 1  //接口绑定
[PPPoE Client-Dialer1]ip ad ppp-negotiate  //通过邻居分配获得IP地址
[PPPoE Client-Dialer1]ppp ipcp dns request  //配置接受DNS服务器

# 配置PAP认证
[PPPoE Client-Dialer1]ppp pap local-user bad password cipher huawei123

# 绑定拨号接口
[PPPoE Client]int g0/0/1
[PPPoE Client-GigabitEthernet0/0/1]pppoe-client dial-bundle-number 1

# 查看被分配的IP地址，进行连通性测试
```

```
[PPPoE Client]ping 100.100.100.254

# 客户端物理接口配置IP地址并配置静态路由
[PPPoE Client]ip route-static 0.0.0.0 0 Dialer 1
[PPPoE Client]int g0/0/0
[PPPoE Client-GigabitEthernet0/0/0]ip ad 192.168.43.254 24

# PPPoE服务器配置静态路由（实际情况中无需配置静态路由）
[PPPoE Server]ip route-static 0.0.0.0 0 100.100.100.253

# PC上连通性测试
```

# DHCP配置

如下拓扑，配置DHCP，使PC1与PC2自动获取IP地址

1. 配置接口地址池
2. 配合全局地址池
3. 配置DHCP，使两台PC获得不同网段的IP地址




## 接口地址池

```
[DHCP Server]dhcp enable  //开启DHCP服务
[DHCP Server]int g0/0/0
[DHCP Server-GigabitEthernet0/0/0]ip ad 192.168.43.254 24  //配置地址
[DHCP Server-GigabitEthernet0/0/0]dhcp select interface  //接口调用
[DHCP Server-GigabitEthernet0/0/0]dhcp server dns-list 8.8.8.8  //配置DNS
[DHCP Server-GigabitEthernet0/0/0]dhcp server excluded-ip-address
192.168.43.244 192.168.43.253  //不参与分配的IP地址
[DHCP Server-GigabitEthernet0/0/0]dhcp server lease day 3  //IP地址租约

# PC使用DHCP获取IP地址，查看IP地址
```

# 全局地址池

```
[DHCP Server]dhcp enable  //开启DHCP服务
[DHCP Server]ip pool bad  //创建全局地址池
[DHCP Server-ip-pool-bad]net 192.168.43.0 mask 24  //添加一个网段
[DHCP Server-ip-pool-bad]gateway-list 192.168.43.254  //配置网关
[DHCP Server-ip-pool-bad]dns-list 114.114.114.114  //配置DNS
[DHCP Server-ip-pool-bad]excluded-ip-address 192.168.43.250 192.168.43.253
//不参与分配的IP地址
[DHCP Server-ip-pool-bad]lease day 5  //IP地址租约时间
[DHCP Server-ip-pool-bad]dis ip pool  //查看地址池的相关信息

# 将接口使用本地地址池
[DHCP Server]int g0/0/0
[DHCP Server-GigabitEthernet0/0/0]dhcp select global  //调用本地的地址池
[DHCP Server-GigabitEthernet0/0/0]ip ad 192.168.43.254 24  //接口添加IP地址
（与地址池的地址同一网段）

# PC查看获取的IP地址
```

- **拓展**：两台PC分配不同网段的IP（此处的配置是继续上面的实验）

**方法一**：配置单臂路由，配置子接口

```
# 交换机上的配置
[SW1]vlan 10
[SW1-vlan10]vlan 20
[SW1-vlan20]int g0/0/2
[SW1-GigabitEthernet0/0/2]port link-type access
[SW1-GigabitEthernet0/0/2]port default vlan 10
[SW1-GigabitEthernet0/0/2]int g0/0/3
[SW1-GigabitEthernet0/0/3]port link-type access
[SW1-GigabitEthernet0/0/3]port default vlan 20
[SW1-GigabitEthernet0/0/3]int g0/0/1
[SW1-GigabitEthernet0/0/1]port link-type trunk
[SW1-GigabitEthernet0/0/1]port trunk allow-pass vlan 10 20

# 路由器上配置
[DHCP Server]int g0/0/0
[DHCP Server-GigabitEthernet0/0/0]undo dhcp select global  //删除DHCP的配置
[DHCP Server-GigabitEthernet0/0/0]undo ip add  //删除IP地址

# 配置子接口
[DHCP Server]int g0/0/0.1
```

```
[DHCP Server-GigabitEthernet0/0/0.1]dot1q termination vid 10  //封装VLAN ID
[DHCP Server-GigabitEthernet0/0/0.1]arp broadcast enable  //开启ARP转发
[DHCP Server-GigabitEthernet0/0/0.1]ip add 192.168.43.254 24  //配置IP地址
[DHCP Server-GigabitEthernet0/0/0.1]int g0/0/0.2
[DHCP Server-GigabitEthernet0/0/0.2]dot1q termination vid 20
[DHCP Server-GigabitEthernet0/0/0.2]arp broadcast enable
[DHCP Server-GigabitEthernet0/0/0.2]ip add 192.168.53.254 24

# 查看地址池
[DHCP Server]dis ip pool

# 创建地址池
[DHCP Server]ip pool boy
[DHCP Server-ip-pool-boy]net 192.168.53.0 mask 24  //分配的网段
[DHCP Server-ip-pool-boy]gateway-list 192.168.53.254  //网关
[DHCP Server-ip-pool-boy]lease day 3  //IP地址租约
[DHCP Server-ip-pool-boy]dns-list 8.8.8.8  //DNS服务器
[DHCP Server-ip-pool-boy]excluded-ip-address 192.168.53.200 192.168.53.253
//不参与分配的IP地址

# 查看地址池
[DHCP Server]dis ip pool

# 接口调用地址池
[DHCP Server]int g0/0/0.1
[DHCP Server-GigabitEthernet0/0/0.1]dhcp select global  //调用全局地址池
[DHCP Server-GigabitEthernet0/0/0.1]int g0/0/0.2
[DHCP Server-GigabitEthernet0/0/0.2]dhcp select global  //调用地址池

# PC查看获取的IP地址
```

- **方法二**：DHCP中继

```
# 配置DHCP中继
[SW1]dhcp enable
[SW1]int Vlanif 10
[SW1-Vlanif10]dhcp select relay
[SW1-Vlanif10]dhcp relay server-ip 192.168.43.254  //DHCP服务器的出接口地址
[SW1-Vlanif10]q
[SW1]int Vlanif 20
[SW1-Vlanif20]dhcp select relay
[SW1-Vlanif20]dhcp relay server-ip 192.168.43.254
```

# AAA

- 配置AAA步骤：
  - 起aaa（`aaa`）
  - 配置本地用户和密码（`local-user bad password cipher huawei@123`）
  - 应用的服务类型（`local-user bad service-type telnet`）
  - 设置权限（`local-user bad privilege level 5`）
  - 允许同时登录的用户数量（`user-interface vty 0 4`）
  - 修改认证模式（`authentication-mode aaa`）

# 配置Telnet和Stelnet登录

```
[AC1]telnet server enable  //开启Telnet服务
[AC1]aaa  //配置aaa
[AC1-aaa]local-user bad password cipher huawei@123  //创建用户并设置密码
[AC1-aaa]local-user bad service-type telnet  //设置账户类型
[AC1-aaa]local-user bad privilege level 5  //设置等级
Warning: This operation may affect online users, are you sure to change the
user privilege level ?[Y/N]y
[AC1-aaa]q

[AC1]user-interface vty 0 4
[AC1-ui-vty0-4]protocol inbound all  // 允许登录接入用户类型的协议
[AC1-ui-vty0-4]authentication-mode aaa  //修改aaa认证模式
[AC1-ui-vty0-4]return

<AC1>telnet 192.168.43.120  //Telnet登录
```

# Stelnet登录

```
# 生本地rsa密钥
[FWQ]rsa local-key-pair create  //创建密钥
The key name will be: Host
% RSA keys defined for Host already exist.
Confirm to replace them? (y/n)[n]:y  //y确认
The range of public key size is (512 ~ 2048).
NOTES: If the key modulus is greater than 512,
       It will take a few minutes.
Input the bits in the modulus[default = 512]:512  //密钥长度
Generating keys...

# 配置AAA认证
[FWQ]aaa
[FWQ-aaa]local-user bad password cipher huawei@123  //创建用户及密码
[FWQ-aaa]local-user bad service-type ssh  //配置用户允许登录方式
[FWQ-aaa]local-user bad privilege level 5  //设置账户等级
[FWQ-aaa]q
[FWQ]user-interface vty 0 4  //配置允许用户登录
[FWQ-ui-vty0-4]authentication-mode aaa  //用户登录的方式
[FWQ-ui-vty0-4]protocol inbound ssh //允许通过ssh登录

# 在系统视图下创建一个用户，指定ssh登录方式为密码登录
[FWQ]ssh user bad authentication-type password  //配置密码登录
[FWQ]stelnet server enable  //开启Stelnet服务
```

- 客户端配置

```
# 开启首次认证
[KH]ssh client first-time enable

# Stelnet登录
[KH]stelnet 2.2.2.29
Please input the username:bad  //用户名
Trying 2.2.2.29 ...
```

```
Press CTRL+K to abort
Connected to 2.2.2.29 ...
The server is not authenticated. Continue to access it? (y/n)[n]:y  //y确认
Apr  2 2020 20:25:28-08:00 KH %%01SSH/4/CONTINUE_KEYEXCHANGE(l)[0]:The
server had not been authenticated in the process of exchanging keys. When
deciding whether to continue, the user chose Y.
[KH]
Save the server's public key? (y/n)[n]:y  //y确认
The server's public key will be saved with the name 2.2.2.29. Please
wait...

Apr  2 2020 20:25:30-08:00 KH %%01SSH/4/SAVE_PUBLICKEY(l)[1]:When deciding
whether to save the server's public key 2.2.2.29, the user chose Y.
[KH]
Enter password:   //密码
<FWQ>
```

# ACL配置

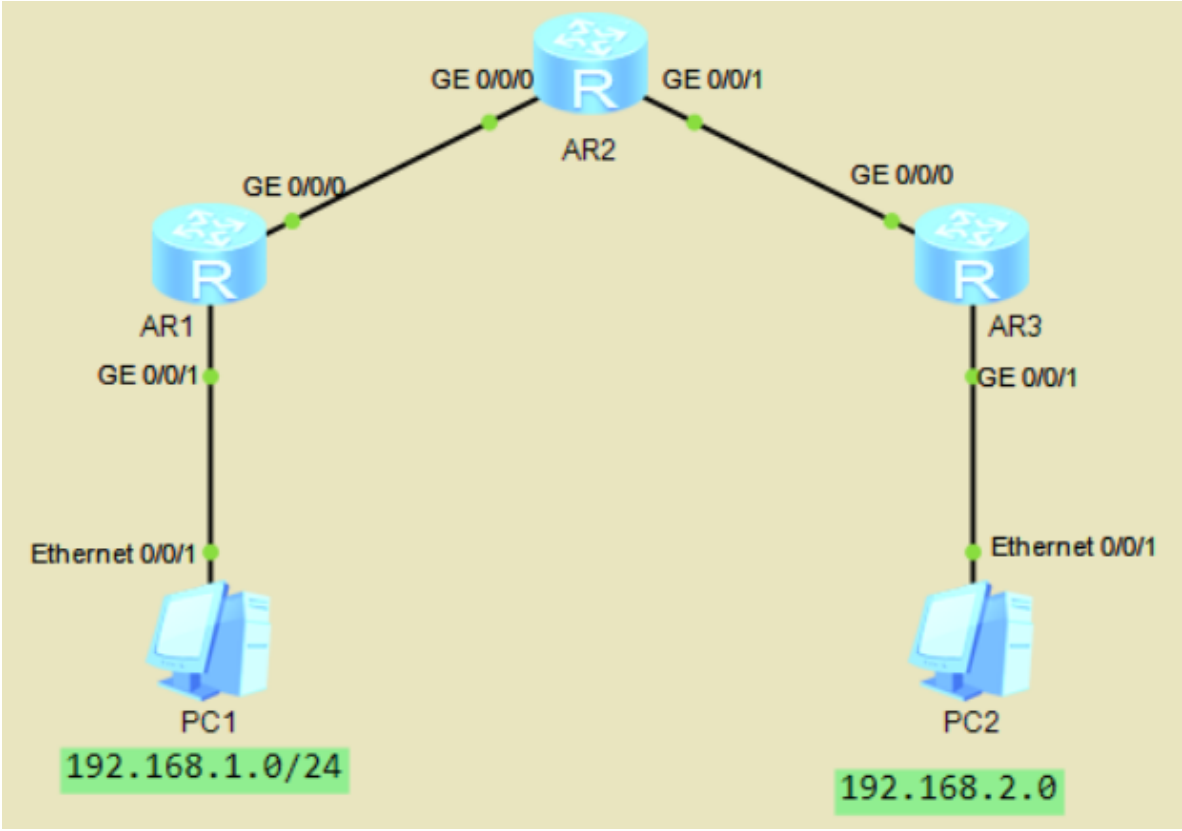
## 基本ACL配置

```
acl 2000
rule deny source 192.168.1.0 0.0.0.255
interface GigabitEthernet 0/0/0
traffic-filter outbound acl 2000  //出方向调用2000规则
```

## 高级ACL配置

```
acl 3000
# 拒绝192.168.1.0网段主机访问172.16.10.1的FTP（21端口）
rule deny tcp source 192.168.1.0 0.0.0.255 destination 172.16.10.1 0.0.0.0
destination-port eq 21
# 拒绝192.168.2.0主机访问172.16.10.2的所有服务
rule deny tcp source 192.168.2.0 0.0.0.255 destination 172.16.10.2 0.0.0.0
rule permit ip  //允许其它，默认为拒绝
traffic-filter outbound acl 3000  //接口出方向调用此ACL
```

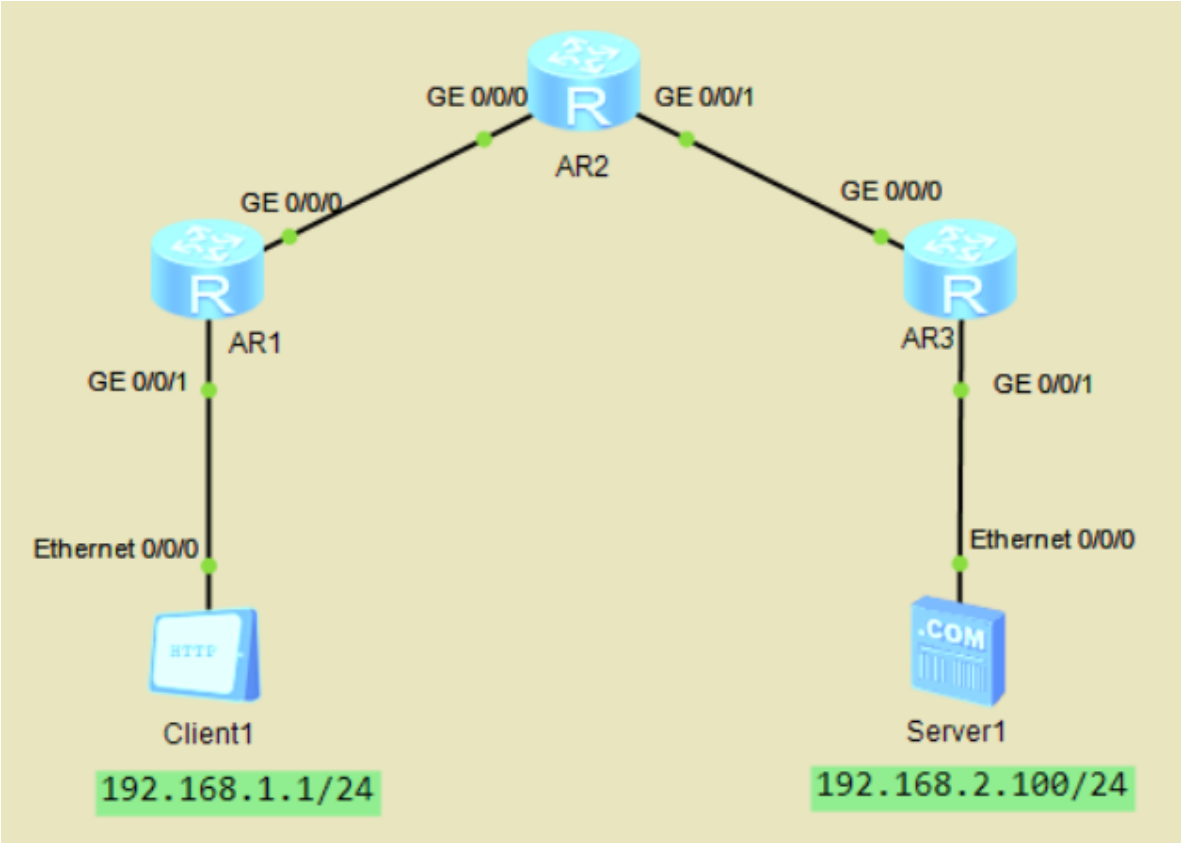
**实验**：如下拓扑图，配置IP地址，配置RIP，使PC间互通，通过配置ACL，阻止PC互通。




- AR2上配置ACL

```
[AR2]acl 2000
[AR2-acl-basic-2000]rule deny source 192.168.1.0 0.0.0.255  //配置ACL
[AR2-acl-basic-2000]rule permit  //放行其他的IP
[AR2-acl-basic-2000]q
[AR2]int g0/0/0
[AR2-GigabitEthernet0/0/0]traffic-filter inbound acl 2000  //接口入方向调用ACL
```

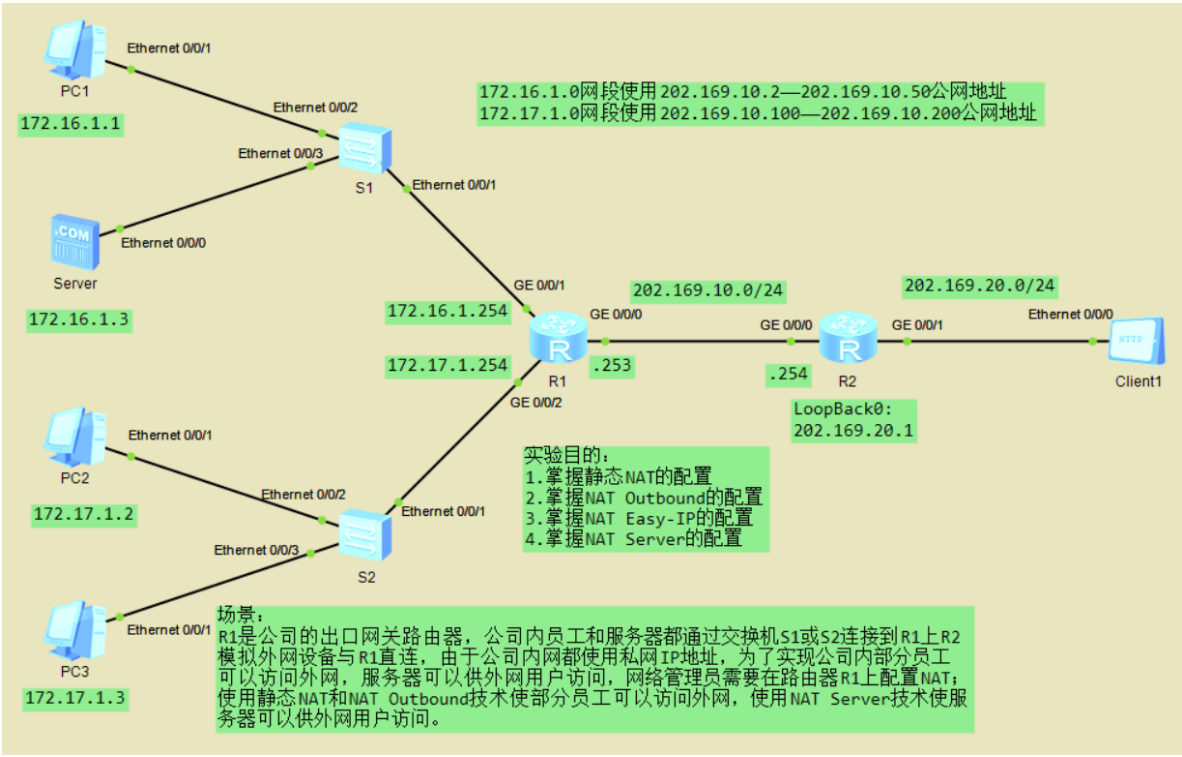
**ACL控制访问FTP服务器**




```
[AR3]acl 3000  //配置ACL
# 禁止192.168.1.0访问192.168.2.100的FTP服务器
[AR3-acl-adv-3000]rule deny tcp source 192.168.1.0 0.0.0.255 destination
192.168.2.100 0 destination-port eq 21
[AR3-acl-adv-3000]q
[AR3]int g0/0/0
[AR3-GigabitEthernet0/0/0]traffic-filter inbound acl 3000  //接口入方向调用ACL
```

# NAT配置

如下拓扑，完成相关IP地址配置，完成相关需求。




# 静态NAT

```
[R1]int g0/0/0
[R1-GigabitEthernet0/0/0]nat static global 202.169.10.3 inside 172.16.1.1  //建立公网地址与私网地址的映射关系
```

# Easy IP

```
# 删除静态NAT
[R1]int g0/0/0
[R1-GigabitEthernet0/0/0]undo nat static global 202.169.10.3 inside 172.16.1.1

# 调用ACL
[R1]acl 2000  //配置ACL
[R1-acl-basic-2000]rule permit  //配置允许所有通过
[R1-acl-basic-2000]q
[R1]int g0/0/0
[R1-GigabitEthernet0/0/0]nat outbound 2000  //接口调用ACL
[R1-GigabitEthernet0/0/0]q
[R1]dis nat outbound  //查看
```

# 动态NAT

```
# 删除Easy IP配置
[R1]int g0/0/0
[R1-GigabitEthernet0/0/0]undo nat outbound 2000
[R1-GigabitEthernet0/0/0]q
[R1]undo acl 2000

# 创建公网地址池
# 创建名为1范围为202.169.10.2-202.169.10.50的地址池
```

```
[R1]nat address-group 1 202.169.10.2 202.169.10.50
# 创建名为2范围为202.169.10.100-202.169.10.200的地址池
[R1]nat address-group 2 202.169.10.100 202.169.10.200

# 配置ACL
[R1]acl 2000
[R1-acl-basic-2000]rule permit source 172.16.1.0 0.0.0.255
[R1-acl-basic-2000]q
[R1]acl 2001
[R1-acl-basic-2001]rule permit source 172.17.1.0 0.0.0.255

# 公网地址池调用ACL
[R1]int g0/0/0
[R1-GigabitEthernet0/0/0]nat outbound 2000 address-group 1 no-pat
[R1-GigabitEthernet0/0/0]nat outbound 2001 address-group 2 no-pat

# 查看地址池
[R1]dis nat outbound
 NAT Outbound Information:
 --------------------------------------------------------------------------
 Interface                       Acl     Address-group/IP/Interface   Type
 --------------------------------------------------------------------------
 GigabitEthernet0/0/0          2000                1               no-pat
 GigabitEthernet0/0/0          2001                2               no-pat
 --------------------------------------------------------------------------
  Total : 2
```

# NAT Server

```
# 删除动态NAT配置
[R1]int g0/0/0
[R1-GigabitEthernet0/0/0]undo nat outbound 2000 address-group 1 no-pat
[R1-GigabitEthernet0/0/0]undo nat outbound 2001 address-group 2 no-pat
[R1-GigabitEthernet0/0/0]q
[R1]undo acl 2000
[R1]undo acl 2001

# 重新配置ACL，并调用
[R1]acl 2000
[R1-acl-basic-2000]rule permit
[R1-acl-basic-2000]q
[R1]int g0/0/0
[R1-GigabitEthernet0/0/0]nat outbound 2000
```

- 配置NAT Server

```
# 配置ftp端口映射
[R1]int g0/0/0
[R1-GigabitEthernet0/0/0]nat server protocol tcp global current-interface ftp
inside 172.16.1.3 ftp
```