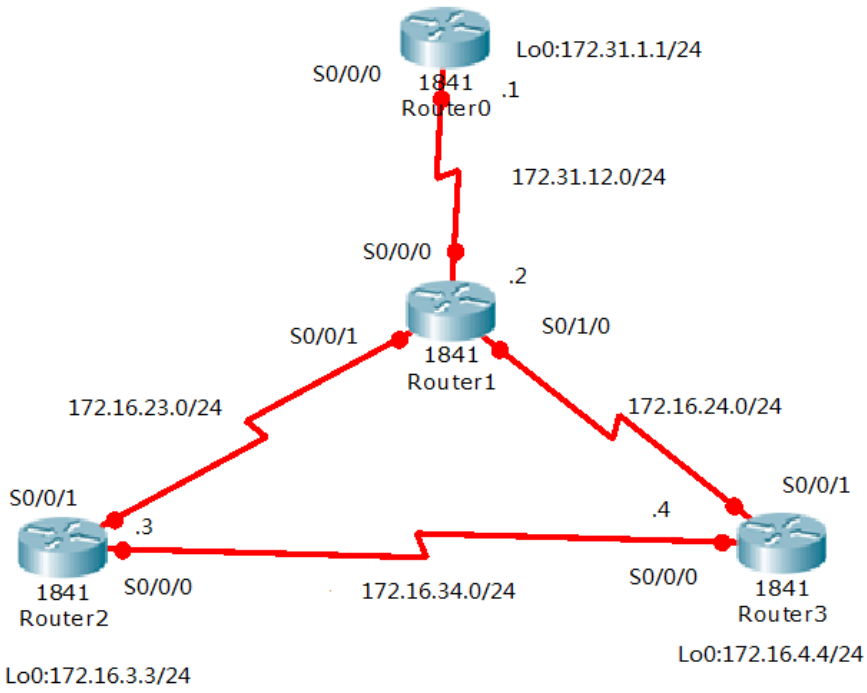
网络路由器配置 RIP 协议

一、 实验内容

1. 实验拓扑图：



2. 为路由器修改名称，熟悉路由器的工作模式。

```
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname 2022ly-1
2022ly-1(config)#
```

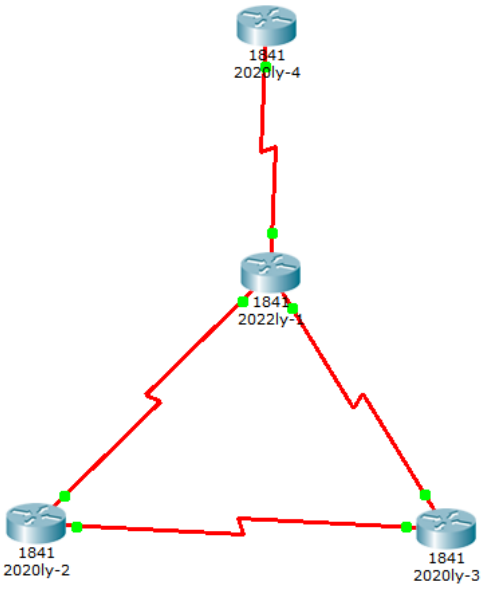
3. 熟悉路由器端口的配置规则，根据实验网络拓扑结构完成路由器端口的基本配置。以中间路由器 router1 的 S0/0/0 配置为例，完成所有端口的配置

```
2022ly-1(config)#interface Serial0/0/0
2022ly-1(config-if)#ip address 172.31.12.2 255.255.255.0
2022ly-1(config-if)#no shutdown

%LINK-5-CHANGED: Interface Serial0/0/0, changed state to down
```

```
2020ly-2(config)#interface lo0

2020ly-2(config-if)#
%LINK-5-CHANGED: Interface Loopback0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Loopback0, changed state to up

2020ly-2(config-if)#ip address 172.16.3.3 255.255.255.0
2020ly-2(config-if)#no shutdown
2020ly-2(config-if)#
```

可以发现此时的 S0/0/0 端口还是 down 状态，需要与他相连的端口也正确配置，才可以呈现“up”状态。

```
2022ly-1(config-if)#
%LINK-5-CHANGED: Interface Serial0/0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/0, changed state to up


%LINK-5-CHANGED: Interface Serial0/0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/0/1, changed state to up


%LINK-5-CHANGED: Interface Serial0/1/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1/0, changed state to up
```



4. 进行 RIPv1 的配置。

   在配置 RIP 之前，查看路由器之前的连通性，1 号路由器只能联通与它直接相连的 2 号路由器端口，2 号路由器的其他端口都无法 ping 通。

```
2022ly-1#ping 172.16.23.3
```
直连端口可以ping通

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.23.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/10/46 ms
```

```
2022ly-1#ping 172.16.34.3
```
非直连端口不可以ping通

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.34.3, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
```

进行 RIPv1 的配置（以中间的 1 号路由器为例）：

```
2022ly-1(config)#router rip
2022ly-1(config-router)#network 172.16.23.0
2022ly-1(config-router)#network 172.16.24.0
2022ly-1(config-router)#network 172.16.12.0
```

查看路由 R0 信息

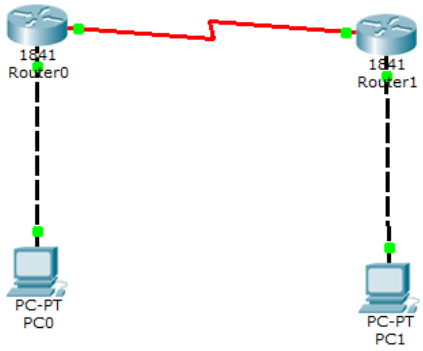
```
2022ly-1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

     172.16.0.0/24 is subnetted, 4 subnets
R       172.16.3.0 [120/1] via 172.16.23.3, 00:00:17, Serial0/0/1
C       172.16.23.0 is directly connected, Serial0/0/1
C       172.16.24.0 is directly connected, Serial0/1/0
R       172.16.34.0 [120/1] via 172.16.23.3, 00:00:17, Serial0/0/1
                    [120/1] via 172.16.24.4, 00:00:16, Serial0/1/0
     172.31.0.0/24 is subnetted, 1 subnets
C       172.31.12.0 is directly connected, Serial0/0/0
```

路由器之间进行 PING 命令，可见所有路由之间都可以 ping 通。

```
2022ly-1#ping 172.16.34.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.34.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/7/29 ms
```

实验一 标准 ACL 实验

(一) 实验拓扑图：



(二) 配置 Telnet 服务

1. 建立超级终端，名称随便取。
2. 选择 COM3 或者 com4，点击还原默认值。
3. 输入 enable,建立 Telnet 的所有命令
4. 在电脑端设置首先要打开电脑的 Telnet 服务
5. 电脑段网卡地址设置。
6. 在菜单栏输入 cmd，打开 DOS 命令框。
7. ping 一下路由器端口地址。
7. 可以 ping 通，之后进行 Telnet 命令。

VTY
术语“ vty ”英文全称为 Virtual teletype，既虚拟终端，用于获取对设备的 Telnet 访问，VTY 仅用于设备的入站连接，这些连接都是虚拟的，没有之联的硬件。抽象的“ 0 - 4 ”表示设备可以同时允许 5 个虚拟连接，可能是 Telnet 或 SSH。
line vty 0 4，该命令是允许用户远程登陆，即不用用户插 Console 线缆，只要设备连接网络，配置了接口 IP 地址即可远程使用 Telnet、或者 ssh 的方式登陆到设备上，，CISCO 设备一般支持 16 个并行的远程虚拟终端，按照编号就是：0 - 15.，Line vty 0 4 就是指同时允许 5 个虚拟终端登陆进行配置,需注意这里配置完成后一定要注意配置 enable 的密码，要不 Telnet 是上不去的。
No shutdown 是开启端口的意思,路由器启机以后端口默认的状态是 shutdown 的,所以必须用 no shutdown 来开启端口

Router0 标准 ACL 列表配置

1. 可以给你的路由器起名字：

Router(config)#hostname Router1

2. 对 router1 的 f0/1 口的地址配置，以及 Telnet 服务地址配置，enable 密码为 501。

```
Router2>
Router2>
Router2>en
Router2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router2(config)#int f0/1
Router2(config-if)#ip address 172.16.1.1 255.255.255.0
Router2(config-if)#exit
Router2(config)#line vty 0 4
Router2(config-line)#password 501
Router2(config-line)#login
Router2(config-line)#exit
Router2(config)#int f0/1
Router2(config-if)#no shutdown
Router2(config-if)#
*Nov 25 01:50:24.459: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state t
o up
*Nov 25 01:50:25.459: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthern
et0/1, changed state to up
Router2(config-if)#exit
Router2(config)#enable password 501
```

3. 对 router1 的 s0/0/0 进行配置。

```
Router2(config-if)#ip address 172.16.12.1 255.255.255.0
Router2(config-if)#no shutdown
Router2(config-if)#exit
Router2(config)#^Z
Router2#sh
*Nov 25 01:54:08.391: %SYS-5-CONFIG I: Configured from console by console
```

4. Ping router2 的 s 口是通的。

```
Router2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router2(config)#int s0/0/0
Router2(config-if)#bandwidth 64
Router2(config-if)#^Z
Router2#
*Nov 25 01:56:41.123: %SYS-5-CONFIG_I: Configured from console by console
Router2#ping 172.16.12.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.12.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 28/28/28 ms
```

5. 在两个路由器全部配置好，PC 配置好后，PC1 是可以 Telnet router2 的
6. 配置 ACL 列表 这里有错误 第二个 Telnet 的 IP 应该是 172.16.0.1

```
Router2(config)#router eigrp 1
Router2(config-router)#network 172.16.1.1 0.0.0.0
Router2(config-router)#network 172.16.2.1 0.0.0.0
Router2(config-router)#network 172.16.12.1 0.0.0.0
Router2(config-router)#no auto-summary
Router2(config-router)#exit
Router2(config)#access-list 2 remark ONLY HOST PC1 CAN TELNET
Router2(config)#access-list 2 permit host 172.16.1.100
Router2(config)#line vty 0 4
Router2(config-line)#access-class 2 in
Router2(config-line)#password 501
Router2(config-line)#login
Router2(config-line)#
*Nov 25 02:00:23.523: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 172.16.12.2 (Se
rial0/0/0) is up: new adjacency
Router2(config-line)#
*Nov 25 02:00:56.683: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 172.16.12.2 (Se
rial0/0/0) is resync: peer graceful-restart
```

配置eigrp协议

定义标准ACL列表，只允许172.16.1.100的Telnet服务访问

应用在VTY线路

7. 本机网卡配置
8. 查看本机的 telnet 服务

EIGRP 是 Cisco 发明的一个私有路由协议,由 IGRP 发展而来,但是算法做了很大的改动.EIGRP 和 IGRP,RIP 一样是一个采用 D-V 算法的动态路由协议

(三) Router1 标准 ACL 列表配置

1. Router2 的 f0/1 以及 s0/0/0 口的配置参照之前的 Telnet 说明文档，f0/1 ip 地址为 172.16.0.1，主机地址为 172.16.0.100，s0/0/0 地址为 172.16.12.2，

```
Router(config)#router eigrp 1
Router(config-router)#network 172.16.12.2 0.0.0.0
Router(config-router)#network 172.16.12.2 0.0.0.0
*Jan  1 00:52:49.659: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 172.16.12.1 (Se
rial0/0/0) is up: new adjacency
Router(config-router)#network 172.16.0.1 0.0.0.0
Router(config-router)#no auto-summary
Router(config-router)#
*Jan  1 00:53:22.787: %DUAL-5-NBRCHANGE: IP-EIGRP(0) 1: Neighbor 172.16.12.1 (Se
rial0/0/0) is resync: summary configuredexit
Router(config)#access-list 2 remark ONLY HOST PC1 CAN TELNET
Router(config)#access-list 2 permit host 172.16.1.100
Router(config)#line vty 0 4
Router(config-line)#access-class 2 in
Router(config-line)#password 501
Router(config-line)#login
Router(config-line)#exit
Router(config)#^Z
```

配置eigrp 路由协议

ACL列表应用于 VTY，也就是telnet 服务中

2. 两个路由器 ACL 列表都配置好后，我们再去看 PC1 是否可以启动路由器 2 的 Telnet 服务
3. 看一下是两个 Telnet 服务
4. 再看 PC2

ip access-group 用在接口下；access-class 用在 VTY 线下
access-class 命令前面没有“ip”

例子：
先配置 access-list：
access-list 1 permit host 192.168.1.1
access-list 1 permit host 192.168.2.1

情况一：line vty 0 4（最多允许 5 个 telnet）
access-class 1 in
情况二：int f0/0
ip access-group 1 in

实验二 扩展 ACL 实验

(一) Router0 扩展 ACL 列表配置

1. 本实验是基于标准 ACL 列表实验基础上进行的，把之前的应用于 vty 的 ACL 标准列表删除。

```
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#line vty 0 4
Router(config-line)#no access-class 2 in
Router(config-line)#^Z
```

2. 删除掉之后，创建新的扩展 ACL 列表 110,该列表最终的目的是拒绝 router2 Telnet router1，即 172.16.1.1，并且不允许 ping 命令应用在两个 PC 之间，即 172.16.0.100 与 172.16.0.100 互相 ping 不通，但是 PC1 可以 Telnet router1 和 router2，即 172.16.1.1 和 172.16.0.1

```
Router(config)#$ 110 deny tcp 172.16.0.0 0.0.0.255 host 172.16.1.1 eq 23
Router(config)#access-list 110 deny icmp 172.16.0.0 0.0.0.255 host 172.16.1.100
Router(config)#access-list 110 permit ip any any
Router(config)#
```

字符过长所以隐藏了，这句话就是拒绝所在网段访问R1的Telnet服务

拒绝PC1与PC2之间进行ping命令

由于ACL列表默认存在deny any命令，所以这一句是必须的

3. 应用列表只应用在 router2 的 s0/0/0 口上。

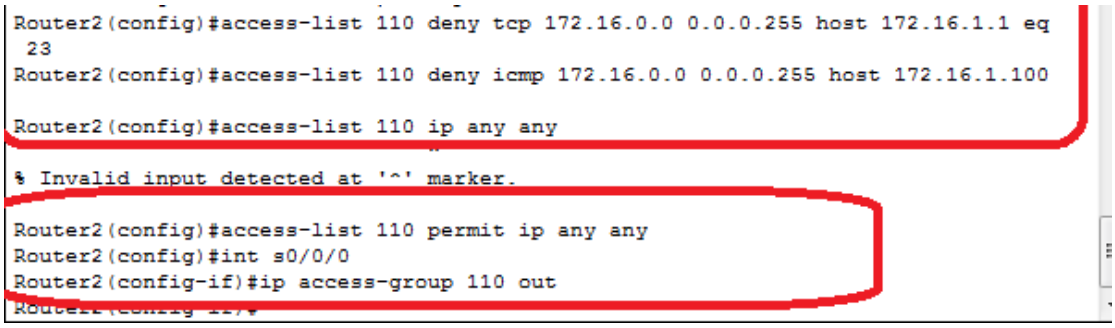
```
Router(config-if)#ip access-group 112 out
```

(二) Router1 扩展 ACL 列表配置

1. 把之前的应用于 vty 的 ACL 标准列表删除

```
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#line vty 0 4
Router(config-line)#no access-class 2 in
Router(config-line)#^Z
```

2. 删除掉之后，创建新的扩展 ACL 列表 110,该列表最终的目的是拒绝 router2 Telnet router1，即 172.16.1.1，并且不允许 ping 命令应用在两个 PC 之间，即 172.16.0.100 与 172.16.0.100 互相 ping 不通，但是 PC1 可以 Telnet router1 和 router2，即 172.16.1.1 和 172.16.0.1，应用列表只应用在 router2 的 s0/0/0 口上。

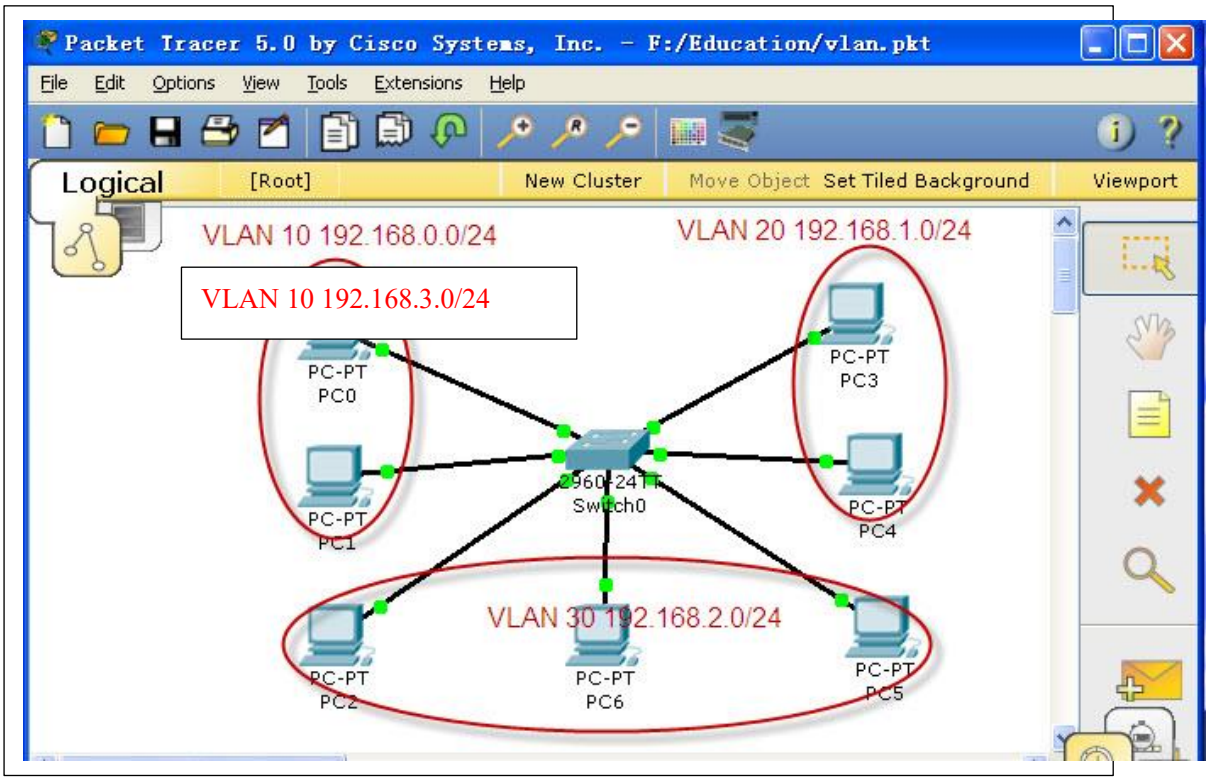


(三) 配置完成后

1. 两个路由器全部配置好后，在 PC2 机上不能 Telnet172.16.1.1，可以 Telnet 172.16.0.1，可以 ping 172.16.1.1，但是不能 ping172.16.1.100。

2. 但是在 PC1 上除了不能 ping 172.16.0.100，之外都可以。

实验一　单一交换机配置 VLAN



一、 实验内容

1. 实验拓扑图
2. 创建 VLAN

在 Cisco IOS 中有两种方式创建 vlan,在全局配置模式下使用 vlan vlanid 命令，如 switch(config)#vlan 10；在 vlan database 下创建 vlan,如 switch(vlan)vlan 20。

```
Switch>en
Switch#conf t
Enter configuration commands, one per line.  End with
Switch(config)#hostname CoreSW
CoreSW(config)#vlan 10
CoreSW(config-vlan)#name Math
CoreSW(config-vlan)#exit
CoreSW(config)#exit
%SYS-5-CONFIG_I: Configured from console by console
CoreSW#vlan database
% Warning: It is recommended to configure VLAN from c
  as VLAN database mode is being deprecated. Please c
  documentation for configuring VTP/VLAN in config mo

CoreSW(vlan)#vlan 20 name Chinese
VLAN 20 added:
    Name: Chinese
CoreSW(vlan)#vlan 30 name Other
VLAN 30 added:
    Name: Other
```

3. 把端口划分给 vlan(基于端口的 vlan)

switch(config)#interface fastethernet0/1　进入端口配置模式

switch(config-if)#switchport mode access　配置端口为 access 模式

switch(config-if)#switchport access vlan 10　把端口划分到 vlan 10

如果一次把多个端口划分给某个 vlan 可以使用 interface range 命令。

```
CoreSW(config-if)#interface range fa0/2 - 4
CoreSW(config-if-range)#switchport mode access
CoreSW(config-if-range)#switchport access vlan 20
CoreSW(config-if-range)#interface range fa0/5 - 6
CoreSW(config-if-range)#switchport mode access
CoreSW(config-if-range)#switchport access vlan 30
CoreSW(config-if-range)#
```

4. 查看 vlan 信息

switch#show vlan

show vlan brief　查看 vlan 简明信息

5. 测试

测试相同 VLAN 间的 pc 可以相互 ping 通，不同 VLAN 间是 PING 不通的。

6. 删除配置

```
CoreSW(config)#interface fa0/8
CoreSW(config-if)#no switchport access vlan 40
CoreSW(config-if)#exit
```

把第 0 个模块中的第 8 个端口从 vlan 40 中删除

```
CoreSW#vlan database
% Warning: It is recommended to configure VLAN from config mode
  as VLAN database mode is being deprecated. Please consult use
  documentation for configuring VTP/VLAN in config mode.

CoreSW(vlan)#no vlan 40
Deleting VLAN 40...
CoreSW(vlan)#
```

实验二　跨交换机配置 VLAN

一、 实验目的和要求

VLAN 是指在一个物理网段内，进行逻辑的划分，划分成若干个虚拟局域网。VLAN 最大的特性是不受物理位置的限制，可以进行灵活的划分。VLAN 具备了一个物理网段所具备的特性。相同 VLAN 内的主机可以相互直接通信，不同 VLAN 间的主机之间互相访问必须经由路由设备进行转发。广播数据包只可以在本 VLAN 内进行广播，不能传输到其他 VLAN 中。

Port VLAN 是实现 VLAN 的方式之一，它利用交换机的端口进行 VLAN 的划分，一个端口只能属于一个 VLAN。

Tag VLAN 是基于交换机端口的另外一种类型，主要用于使交换机的相同 VLAN 内的主机之间可以直接访问，同时对于不同 VLAN 的主机进行隔离。Tag VLAN 遵循 IEEE802.1Q 协议的标准。在使用配置了 Tag VLAN 的端口进行数据传输时，需要在数据帧内添加 4 个字节的 802.1Q 标签信息，用于标示该数据帧属于哪个 VLAN，便于对端交换机收到数据帧后进行准确的过滤。

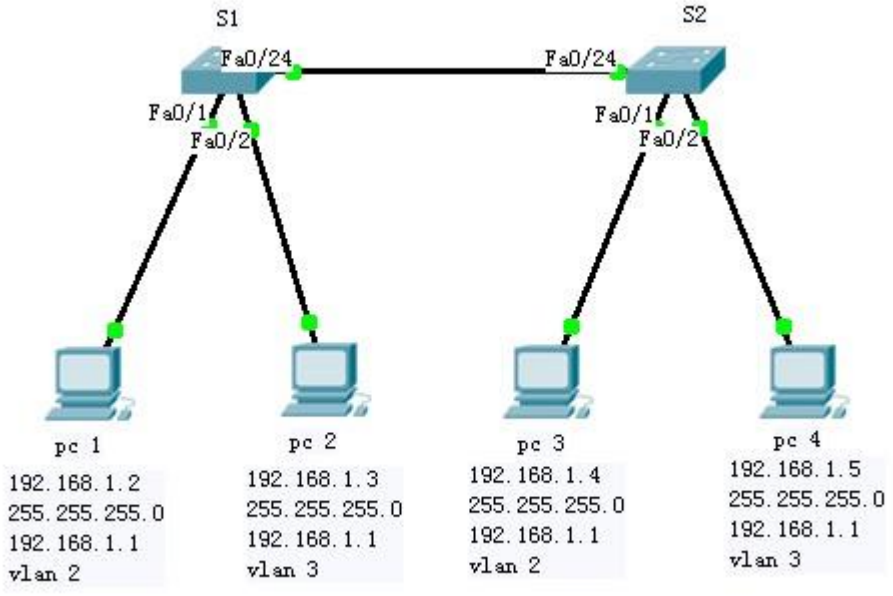
要求：1.理解虚拟 LAN(VLAN)基本原理；

2.掌握一般交换机按端口划分 VLAN 的配置方法；

3 .掌握 Tag VLAN 配置方法。。

二、 实验内容

(一)　实验拓扑



(二) 配置 VLAN

1. 设置四台 PC 机 IP 地址。
2. 对交换机 S1 进行设置。

```
S1(config)#vlan 2                                    //划分 VLAN 2
S1(config-vlan)#exit
S1(config)#vlan 3                                    //划分 VLAN 3
S1(config-vlan)#exit
S1(config)#interface fa0/1
S1(config-if)#switchport mode access
S1(config-if)#switchport access vlan 2          //将 fa0/1 划分到 VLAN 2
S1(config-if)#exit
S1(config)#interface fa0/2
S1(config-if)#switchport mode access
S1(config-if)#switchport access vlan 3          //将 fa0/2 划分到 VLAN 3
S1(config-if)#exit
S1(config)#interface fa0/24                     //设置 fa0/24 端口模式为 trunk
S1(config-if)#switchport mode trunk
S1(config-if)#end
```

3. 查看 S1 的 VLAN 状况
4. 对交换机 S2 进行 VLAN 设置。

```
S2(config)#vlan 2
S2(config-vlan)#exit
S2(config)#vlan 3
S2(config-vlan)#exit
S2(config)#interface fa0/1
S1(config-if)#switchport mode access
S2(config-if)#switchport access vlan 2
S2(config-if)#exit
S2(config)#interface fa0/2
S1(config-if)#switchport mode access
S2(config-if)#switchport access vlan 3
S2(config-if)#exit
S2(config)#interface fa0/24
S2(config-if)#switchport mode trunk
S2(config-if)#end
```

实验三 单臂路由配置

一 实验原理

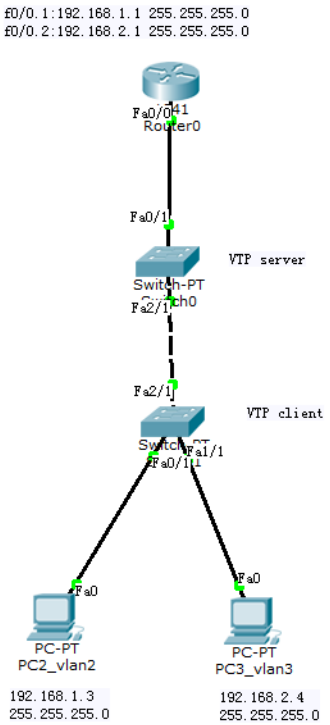
本实验接上一个实验，计算机和交换机的 IP 地址和网关不变，但要求交换机工作在两个 VLAN 的情况下，新创建的 vlan，一个是 vlan 2，另一个是 vlan 3. 当交换机设置成两个 vlan 时，逻辑上已经成为两个网络，广播被隔离了。两个 vlan 的网络要通信，必须通过路由器，如果接入路由器的一个物理端口，则必须有两个子接口分别与两个 vlan 对应，同时还要求与路由器相联的交换机的端口 f0/1 要设置为 trunk，因为这个口要通过两个 vlan 的数据包。

二、实验目的：

通过实验掌握单臂理由的原理及应用，并且熟悉上次试验 vtp 原理。

三、实验步骤：

(1) 试验拓扑图如下：



(2)把 Fa0/1 设为 trunk 模式 交换机之间相连的端口设为 trunk 模式

a. 交换机 0 VTP server

```
Switch(vlan)#vtp server
Device mode already VTP SERVER.
Switch(vlan)#vtp domain test
Changing VTP domain name from NULL to test
Switch(vlan)#vtp password 123
Setting device VLAN database password to 123
Switch(vlan)#exit
APPLY completed.
Exiting....
Switch#show vtp status
VTP Version                     : 2
Configuration Revision          : 7
Maximum VLANs supported locally : 255
Number of existing VLANs        : 7
VTP Operating Mode              : Server
VTP Domain Name                 : test
VTP Pruning Mode                : Disabled
VTP V2 Mode                     : Disabled
VTP Traps Generation            : Disabled
MD5 digest                      : 0xDC 0x3B 0x17 0x69 0xD8 0x3E 0xAE 0x9B
Configuration last modified by 0.0.0.0 at 3-1-93 00:05:43
Local updater ID is 0.0.0.0 (no valid interface found)
```

b．交换机 1 VTP client

```
Switch(vlan)#vtp client
Setting device to VTP CLIENT mode.
Switch(vlan)#vtp domain test
Domain name already set to test.
Switch(vlan)#vtp server 123
                        ^
% Invalid input detected at '^' marker.

Switch(vlan)#vtp password 123
Setting device VLAN database password to 123
Switch(vlan)#exit
APPLY completed.
Exiting....
Switch#sh vtp status
VTP Version                     : 2
Configuration Revision          : 7
Maximum VLANs supported locally : 255
Number of existing VLANs        : 7
VTP Operating Mode              : Client
VTP Domain Name                 : test
VTP Pruning Mode                : Disabled
VTP V2 Mode                     : Disabled
VTP Traps Generation            : Disabled
MD5 digest                      : 0xDC 0x3B 0x17 0x69 0xD8 0x3E 0xAE 0x9B
Configuration last modified by 0.0.0.0 at 3-1-93 00:05:43
```

（3）在 Client 交换机上，将 PC 添加到对应 vlan 中：

a．把 PC2 添加到 vlan 2：

Switch(config)#interface FastEthernet0/1

Switch(config-if)#switchport access vlan 2

b．PC3 添加到 vlan 3：

（4）将交换机 0 与路由器相联的端口 f0/1 要设置为 trunk：

(5)路由器的配置：

a、激活路由器与交换机相连的端口，b、进入该端口的子接口，c、封装 dot1Q 协议，d、添加子接口的 ip 地址

```
Router>ena
Router#configure terminal
Enter configuration commands, one per line.   End with CNTL/Z.
Router(config)#int f0/0.1
Router(config-subif)#encapsulation dot1Q 2
Router(config-subif)#ip address 192.168.1.1 255.255.255.0
Router(config-subif)#no shut
Router(config-subif)#exit
Router(config)#int f0/0.2
Router(config-subif)#encapsulation dot1Q 3
Router(config-subif)#ip address 192.168.2.1 255.255.255.0
Router(config-subif)#no shut
Router(config-subif)#exit
Router(config)#exit
Router#
```

五、实验结果

经过单臂路由的配置，可以将不同VLAN下的PC相互连通。

六、实验总结

单臂路由是为了节约接口而实现不同 vlan 间通信的一种技术，关键在于子接口的配置。由于数据流经过 trunk 口的时候是不会解封数据的 vlan 标签的，而路由器的物理接口又不识别携带了 vlan 标签的数据帧，因此使用逻辑子接口，封装 dot1Q 协议，比如：encapsulation dot1q 2，这样，子接口便能识别携带了 vlan2 标签的数据帧，然后路由器可以查路由表，从封装了相应 dot1q 标签的子接口将数据转发出去，从而实现了不同 vlan 间的通信，但子接口共用一个物理接口限制了带宽。在配置过程中有过实验失败的经历，经仔细检查发现，可能是在进入子接口配置后，未输入“Router(config-subif)#no shut”将子接口激活，也可能是在交换机之间的接口为配置成 ACCESS。最终还是成功完成了实验。

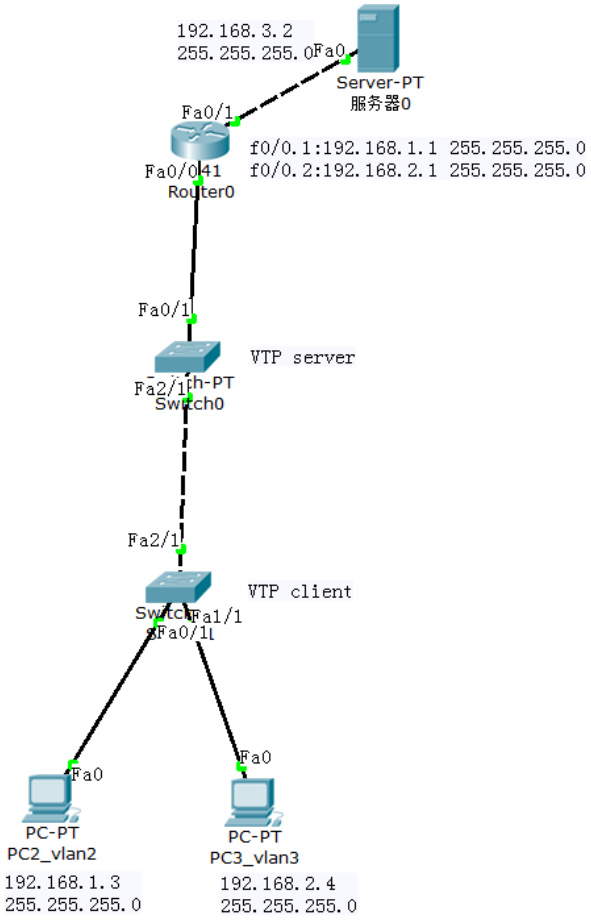
实验四 路由器的 NAT---PAT 配置

一、实验原理

PAT(Port Address Translate)是一种特殊动态 NAT，它用于将多个内部本地 IP 地址映射到一个公网 IP 的不同端口上。将原动态 nat 命令行地址池 pool 改变成为对外接口 s0/0，并在后边加上参数 overload。

三、实验步骤

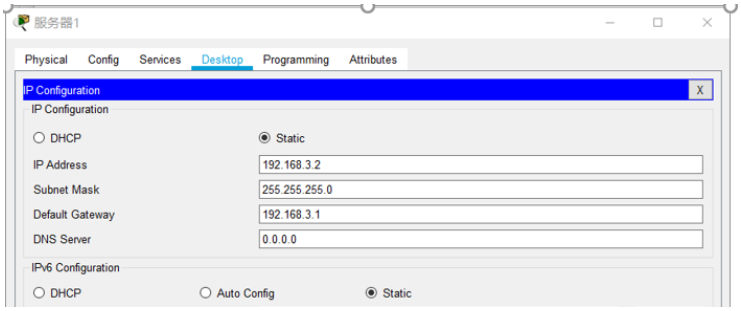
（1） 实验拓扑图



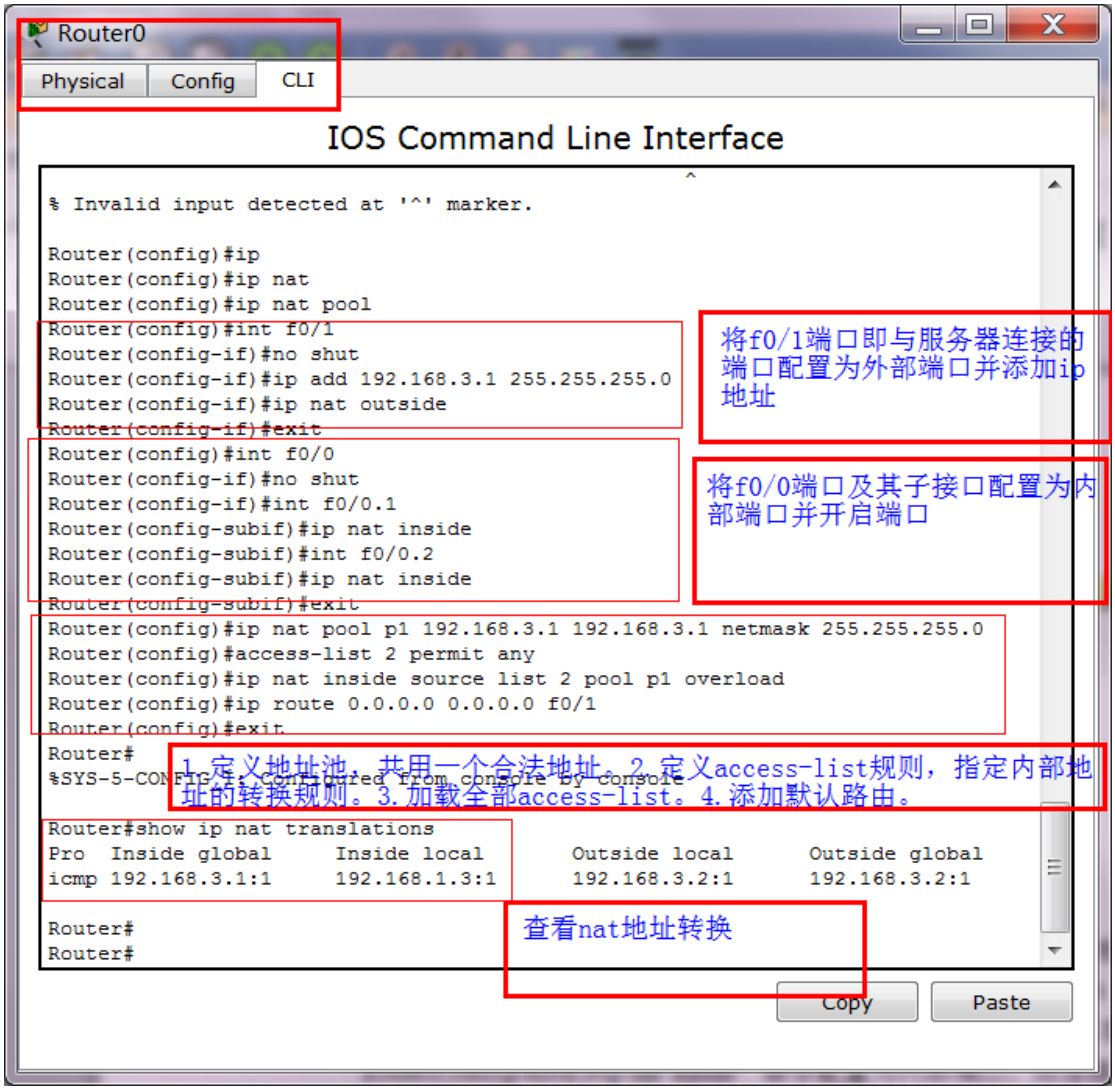
（2） 在上次实验的基础上添加一个服务器并和路由器连接，开启服务器和路由器的端口 f1/0 添加 ip

```
Router(config)#interface FastEthernet0/1
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
ip address 192.168.3.1 255.255.255.0
Router(config-if)#ip address 192.168.3.1 255.255.255.0
```

（3）为服务器添加 ip



（3） 路由器的 NAT—PAT 的配置



四、实验总结

Nat 转换是将内部地址映射为公网 IP 地址，可以方便解决局域网内大量 IP 地址的申请问题，Nat 转换节约了有限的 IPv4 的地址，屏蔽了内部网络细节，有一定的安全作用。实验关键步骤在于路由器的端口配置，比如 f0/0 需要用“Router(config-subif)#ip nat inside”命令配置为内部端口，而连接外部服务器的端口需配置为外部端口。最后启用 nat 服务以及加载的 access-list 规则实现路由器的 nat 转换配置

实验五 三层交换机的配置

一、三层交换的概念

交换机是链路层设备，使用 MAC 地址，完成对帧的操作。交换机的 IP 地址做管理用，交换机的 IP 地址实际是 VALN 的 IP。一个 VLAN 一个广播域，不同 VLAN 的主机间访问，相当于网络间的访问，要通过路由实现。

不同 VLAN 间主机的访问有以下几种情况：

(1)两个 VLAN 分别接入路由器的两个物理接口。这是路由器的基本应用。

(2)两个 VLAN 通过 trunk 接入路由器的一个物理接口，这 是应用于子接口的单臂路由。

1)通过 VLAN 的 IP 地址做网关，实现三层交换，要求设置 VLAN 的 IP 地址。

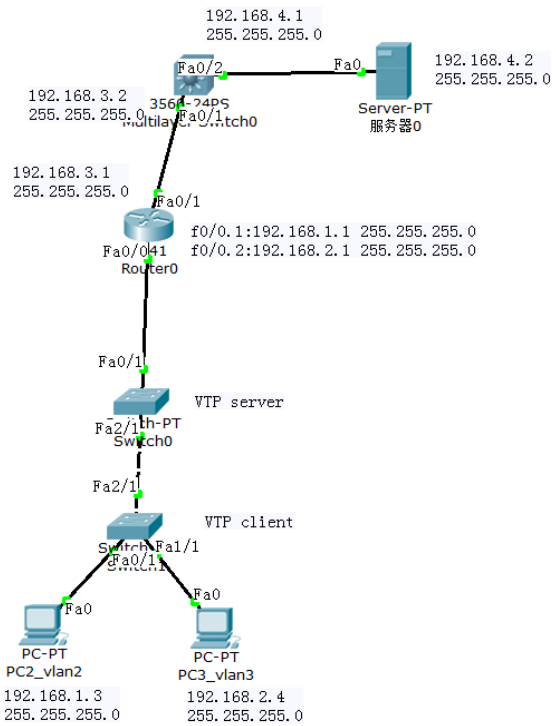
2)将端口设置在三层工作，要求端口设置 no switchport，再设置端口的 IP 地址。

二、试验目的

通过试验掌握三层交换工作原理。

三、实验步骤

（1） 实验拓扑图



（2）给三层交换机创建两个 vlan：vlan2 和 vlan3

（3）把三层交换机的端口添加到相应 vlan

A．将 f0/1 添加到 vlan 2

B．将 f0/2 添加到 vlan 3

（4）给 vlan2 和 vlan 3 添加 IP 地址

```
Switch(config-if)#int vlan 2
Switch(config-if)#
%LINK-5-CHANGED: Interface Vlan2, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan2, changed
state to up

Switch(config-if)#ip add 192.168.3.2 255.255.255.0
Switch(config-if)#no shut
Switch(config-if)#int vlan 3
Switch(config-if)#
%LINK-5-CHANGED: Interface Vlan3, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan3, changed
state to up
ip add 192.168.4.1 255.255.255.0
Switch(config-if)#no shut
```

（6）在交换机上添加 rip 路由协议

```
Switch(config)#router rip
IP routing not enabled
Switch(config)#ip routing
Switch(config)#router rip
Switch(config-router)#network 192.168.3.0
Switch(config-router)#network 192.168.4.0
Switch(config-router)#
```

四、测试连通性

在 pc0 点击“桌面”， 打开 WEB 浏览器 输入服务器的地址

五、实验总结

三层交换机是具有部分路由器功能的交换机，三层交换机的最重要目的是加快大型局域网内部的数据交换，路由功能也是为这目的服务的，能够做到一次路由，多次转发。由硬件高速实现对数据包转发等。在前几次实验成功的基础上，这次试验比较顺利，这次配置的关键在与三层交换机的连接两个不同网络的端口配置，需要给不同网络划分为不同 VLAN，将对应端口添加到不同 vlan，并给端口添加 ip 地址。最后需要交换机添加 RIP 协议并开启路由功能。

实验一 利用 wireshark 分析 HTTP 协议

一、 实验步骤

1. 利用 Wireshark 俘获 HTTP 分组

1) 在进行跟踪之前，我们首先清空 Web 浏览器的高速缓存来确保 Web 网页是从网络中获取的，而不是从高速缓冲中取得的。之后，还要在客户端清空 DNS 高速缓存，来确保 Web 服务器域名到 IP 地址的映射是从网络中请求。在 WindowsXP 机器上，可在命令提示行输入 ipconfig/flushdns 完成操作。
2) 启动 Wireshrk 分组俘获器。
3) 在 Web 浏览器中输入：http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html
4) 停止分组捕获。

浏览 Web 页面经过如下三个过程：

i. DNS 解析

在 URL http://gaia.cs.umass.edu 中，gaia.cs.umass.edu 是一个具体的 web 服务器的域名。最前面有两个 DNS 分组。第一个分组是将域名 gaia.cs.umass.edu 转换成为对应的 IP 地址的请求，第二个分组包含了转换的结果。这个转换是必要的，因为网络层协议——IP 协议，是通过点分十进制来表示因特网主机的，而不是通过 gaia.cs.umass.edu 这样的域名。当输入 URL http://gaia.cs.umass.edu 时，将要求 Web 服务器从主机 gaia.cs.umass.edu 上请求数据，但首先 Web 浏览器必须确定这个主机的 IP 地址。

ii. TCP 连接建立

随着转换的完成，Web 浏览器与 Web 服务器建立一个 TCP 连接。

iii. HTTP 交互

Web 浏览器使用已建立好的 TCP 连接来发送请求“GET/HTTP/1.1”。这个分组描述了要求的行为（“GET”）及文件（只写“/”是因为我们没有指定额外的文件名），还有所用到的协议的版本（“HTTP/1.1”）。

1)DNS 协议主要使用 UDP 封装还是 TCP 封装？

__TCP 封装____

2) DNS 查询消息的 IP 地址是___221.204.7.205_____？你默认的本地 DNS 服务器的 IP 地址是___10.30.58.15_____？

3)DNS 查询消息的目的端口是___domain(53)_____？DNS *响应消息*的源端口是____20430_____？

2、HTTP GET/response 交互

（1）在协议框中，选择“GET/HTTP/1.1” 所在的分组会看到这个基本请求行后跟随着一系列额外的请求首部。在首部后的“\r\n”表示一个回车和换行，以此将该首部与下一个首部隔开。

“Host”首部在 HTTP1.1 版本中是必须的，它描述了 URL 中机器的域名，本例中是 http://gaia.cs.umass.edu。这就允许了一个 Web 服务器在同一时间支持许多不同的域名。有了这个数不，Web 服务器就可以区别客户试图连接哪一个 Web 服务器，并对每个客户响应不同的内容，这就是 HTTP1.0 到 1.1 版本的主要变化。

User-Agent 首部描述了提出请求的 Web 浏览器及客户机器。

接下来是一系列的 Accpet 首部，包括 Accept（接受）、Accept-Language（接受语言）、Accept-Encoding（接受编码）、Accept-Charset（接受字符集）。它们告诉 Web 服务器客户 Web 浏览器准备处理的数据类型。Web 服务器可以将数据转变为不同的语言和格式。这些首部表明了客户的能力和偏好。

Keep-Alive 及 Connection 首部描述了有关 TCP 连接的信息，通过此连接发送 HTTP 请求和响应。它表明在发送请求之后连接是否保持活动状态及保持多久。大多数 HTTP1.1 连接是持久的（persistent），意思是在每次请求后不关闭 TCP 连接，而是保持该连接以接受从同一台服务器发来的多个请求。

（2）我们已经察看了由 Web 浏览器发送的请求，现在我们来观察 Web 服务器的回答。响应首先发送“HTTP/1.1 200 ok”，指明它开始使用 HTTP1.1 版本来发送网页。同样，在响应分组中，它后面也跟随着一些首部。最后，被请求的实际数据被发送。

第一个 Cache-control 首部，用于描述是否将数据的副本存储或高速缓存起来，以便将来引用。一般个人的 Web 浏览器会高速缓存一些本机最近访问过的网页，随后对同一页面再次进行访问时，如果该网页仍存储于高速缓存中，则不再向服务器请求数据。类似地，在同一个网络中的计算机可以共享一些存在高速缓存中的页面，防止多个用户通过到其他网路的低速网路连接从网上获取相同的数据。这样的高速缓存被称为代理高速缓存（proxy cache）。在我们所俘获的分组中我们看到“Cache-control”首部值是“private”的。这表明服务器已经对这个用户产生了一个个性化的响应，而且可以被存储在本地的高速缓存中，但不是共享的高速缓存代理。

在 HTTP 请求中，Web 服务器列出内容类型及可接受的内容编码。此例中 Web 服务器选择发送内容的类型是 text/html 且内容编码是 gzip。这表明数据部分是压缩了的 HTML。

服务器描述了一些关于自身的信息。此例中，Web 服务器软件是 Google 自己的 Web 服务器软件。响应分组还用 Content-Length 首部描述了数据的长度。最后，服务器还在 Date 首部中列出了数据发送的日期和时间。

根据俘获窗口内容，回答 1-6 题。

（1）你的浏览器运行的是 HTTP1.0，还是 HTTP1.1？你所访问的服务器所运行的 HTTP 版本号是多少？

HTTP1.1　　version1.1

（2）你的浏览器向服务器指出它能接收何种语言版本的对象？

Accept-Language:zh-cn\r\n

（3）你的计算机的 IP 地址是多少？

10.30.58.15

（4）从服务器向你的浏览器返回的状态代码是多少？

200ok

（5）你从服务器上所获取的 HTML 文件的最后修改时间是多少？

Date: Wed, 30 Apr 2014 08:16:13 GMT\r\n

（6）返回到你的浏览器的内容一共多少字节？

52031

3、HTTP 条件 GET/response 交互

（1）启动浏览器，清空浏览器的缓存。(在ie:工具----Internet选项-----删除----Internet临时文件 历史记录 删除缓存中的内容)

（2）启动Wireshark分组俘获器，开始Wireshark分组俘获。

（3）在浏览器地址栏中如下网址：

http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html

你的浏览器中将显示一个具有五行的非常简单的 HTML 文件。

（4）在你的浏览器中重新输入相同的 URL 或单击浏览器中的“刷新”按钮。

（5）停止Wireshark分组俘获，在显示过滤筛选说明处输入“http”,分组列表子窗口中将只显示所俘获到的HTTP报文。

根据操作回答 7-10 题。

（7）分析你的浏览器向服务器发出的第一个 HTTP GET 请求的内容，在该请求报文中，是否有一行是：IF-MODIFIED-SINCE?

没有

（8）分析服务器响应报文的内容，服务器是否明确返回了文件的内容？如何获知？ Congratulations again! Now you've downloaded the file lab2-2.html.
This file's last modification date will not change.

Thus if you download this multiple times on your browser, a complete copy will only be sent once by the server due to the inclusion of the IN-MODIFIED-SINCE
field in your browser's HTTP GET request to the server.

（9）分析你的浏览器向服务器发出的第二个“HTTP GET”请求，在该请求报文中是否有一行是：IF-MODIFIED-SINCE？如果有，在该首部行后面跟着的信息是什么？ 没有

（10）服务器对第二个 HTTP GET 请求的响应中的 HTTP 状态代码是多少？服务器是否明确返回了文件的内容？请解释

状态码和相应状态信息的值为 304 NOT Modified,他表示缓存器可以使用该对象。 第二次没有返回文件的内容，因为他只是作为对该条件 GET 的响应，WEB 服务器只发送一个响应报文，不包含请求的对象

4、获取长文件

（1）启动浏览器，将浏览器的缓存清空。

（2）启动 Wireshark 分组俘获器，开始 Wireshark 分组俘获。

（3）在浏览器地址栏中输入如下网址:

http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file3.html

浏览器将显示一个相当大的美国权力法案

（4）停止 Wireshark 分组俘获，在显示过滤筛选说明处输入“http”,分组列表子窗口中将只显示所俘获到的 HTTP 报文。

根据操作回答 11-14 题。

（11）你的浏览器一共发出了多少个 HTTP GET 请求？_____1 个
（12）查看 GET 的响应消息，该单个 HTTP 响应消息，需要多少个 TCP 报文段？
_____4 个
（13）与这个 HTTP GET 请求相对应的响应报文的状态代码和状态短语是什么
_HTTP/1.1 200OK\r___

5、嵌有对象的 HTML 文档

（1）启动浏览器，将浏览器的缓存清空。

（2）启动 Wireshark 分组俘获器。开始 Wireshark 分组俘获。

（3）在浏览器地址栏中输入如下网址：

http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file4.html

浏览器将显示一个具有两个图片的短HTTP文件。

（4）停止 Wireshark 分组俘获，在显示过滤筛选说明处输入“http”,分组列表子窗口中将只显示所俘获到的 HTTP 报文。

根据操作回答 15-16 题。

（15）你的浏览器一共发出了多少个 HTTP GET 请求？这些请求被发送到的目的地的 IP 地址是多少？_3 个__222.198.2.63_

（16）浏览器在下载这两个图片时，是串行下载还是并行下载？请解释。
__当前一个 GET 还没有得到回复就发送了下一个 GET

6、HTTP 认证

（1）启动浏览器，将浏览器的缓存清空。

（2）启动 Wireshark 分组俘获器。开始 Wireshark 分组俘获。

（3）在浏览器地址栏中输入如下网址：

http://gaia.cs.umass.edu/wireshark-labs/protected_pages/HTTP-wireshark-file5.html

浏览器将显示一个HTTP文件，输入所需要的用户名和密码(用户名：wireshark-students,密码:network)。

（4）停止 Wireshark 分组俘获，在显示过滤筛选说明处输入“http”,分组列表子窗口中将只显示所俘获到的 HTTP 报文。

根据操作回答 17-18 题。

（17）对于浏览器发出的最初的 HTTP GET 请求，服务器的响应是什么(状态代码和状态短语)?
２００ＯＫ

（18）当浏览器发出第二个 HTTP GET 请求时，在 HTTP GET 响应报文中包含了哪些新的字段？
Authorization :basic

实验二 利用 Wireshark 分析 ICMP 协议

(一) ICMP 协议分析

步骤 1：在 PC1 运行 Wireshark，开始截获报文，为了只截获和实验内容有关的报文，将 Wireshark 的 Captrue Filter 设置为“No Broadcast and no Multicast”；

步骤 2：在 PC1 以 www.sina.com.cn 为目标主机，在命令行窗口执行 Ping 命令，要求 ping 通 10 次；

Ping 命令为：ping – n 10 www.sina.com.cn

步骤 3：停止截获报文，分析截获的结果，回答下列问题：

1）将抓包结果进行截图（要求只显示 ping 的数据包）：

2）截获的 ICMP 报文有几种类型？分别是：两种类型 0 和 8

3）分析截获的 ICMP 报文，

4）查看ping请求分组，ICMP的type是 8 和code是 0

5）查看相应得ICMP响应信息，ICMP的type是 0 和code是 0

6）若要只显示ICMP的echo响应数据包，显示过滤器的规则为 icmp.type==0 并根据过滤规则进行抓包截图

7）若要只显示ICMP的echo请求数据包，显示过滤器的规则为 icmp.type==8 并根据过滤规则进行抓包截图

(二) ICMP 和 Traceroute

在Wireshark 下，用Traceroute程序俘获ICMP分组。Traceroute能够映射出通往特定的因特网主机途径的所有中间主机。

源端发送一串ICMP分组到目的端。发送的第一个分组时，TTL=1；发送第二个分组时，TTL=2，依次类推。路由器把经过它的每一个分组TTL字段值减1。当一个分组到达了路由器时的TTL字段为1时，路由器会发送一个ICMP错误分组（ICMP error packet）给源端。

步骤 4：在 PC1 上运行 Wireshark 开始截获报文；

步骤 5：在 PC1 上执行 Tracert 命令，如：Tracert www.sina.com.cn；将命令窗口进行截图。

设置显示过滤器为 icmp，图 5 显示的是一个路由器返回的 ICMP 超时报告分组（ICMP error packet）。注意到 ICMP 超时报告分组中包括的信息比 Ping ICMP 中超时报告分组包含的信息多。

实验三 利用 Wireshark 分析 tcp 协议

一、 实验原理和要求

TCP 协议工作原理参考 TCP 协议

Tcp 显示过滤规则:

tcp.flags 显示包含 TCP 标志的封包。

tcp.flags.syn == 1 显示包含 TCP SYN 标志的封包。

tcp.flags.syn == 1 and tcp.flags.ack == 0 显示包含 TCP SYN 并且不包含 ACK 标志的封包。

tcp.flags.fin == 1 and tcp.flags.ack == 1 显示包含 TCP FIN 和 ACK 标志的封包。

tcp.window_size == 0 && tcp.flags.reset != 1

要求：

1）掌握 TCP 连接建立的三次握手过程

2）理解 TCP 连接释放的四次握手过程

二、 实验内容

1） 启动 WireShark 抓包

2） 访问学校主页服务器 ，通过 Wireshark 捕获通信内容

3） 分析 TCP 连接建立的三次握手和连接释放的四次握手过程

填写各项的信息及作用

TCP 数据报中依次包括以下信息：

1、Source Port：1101 ，表示 目标通过 1101 端口传送数据 。该部分占 16 个 BIT。

2、Destination Port：80 ，表示 本机通过 80 接口接收数据 。该部分占 16 个 BIT。

3、Initial Sequence Number：584，表示 每一个字节的编号 ，即 SEQ 值。该部分占 32 个 BIT，值从 1 到 2 的 32 次方减 1。

4、Next Expected SEQ Numbe：2214，表示 下一个要接到数据的编号，即对方返回的 ACK 值。该部分占 32 个 BIT，值从 1 到 2 的 32 次方减 1。

5、Data Offset： 4 Bytes，表示 报头大小 。该部分占 32 个 BIT。

6、Reserved Bites：保留位，此处不用。该部分占 6 个 BIT。

7、Flags： 0x0010 。该值用两个十六进制数来表示。该部分长度为 3 个

BIT，6 个标志位的含义分别是：
URG:　0　，为 1 表示　紧急指针有效　。
ACK:　1　，为 1 表示　确认报文　。
PSH:　0　，为 1 表示　发送报文　。
RST:　0　。为 1 表示　TCP 出现严重错误必须释放　
SYN:　0　。为 1 表示　一个连接请求　。
FIN:　0　。为 1 表示　随访发送数据已完毕　。
8、Window：　0　，表示　接收方目前允许对方发送的数据量　。该部分占　16　个 BIT。
9、CheckSum：　0xae1c，表示　校验和　。该部分占　16　个 BIT，用十六进制表示。
10、Urgent Pointer：　00　，表示　紧急数据的字节数　。该部分占　2　个 BIT。
11、Maximum Segment Size：　1514　，表示　数据字段的最大长度　。
【思考问题】
1.试用具体例子说明为什么传输连接建立时要使用三次握手。如不这样做可能会出现什么情况。
答：我们知道，3 次握手完成两个重要的功能，既要双方做好发送数据的准备工作（双方都知道彼此已准备好），也要允许双方就初始序列号进行协商，这个序列号在握手过程中被发送和确认。
现在把三次握手改成仅需要两次握手，死锁是可能发生的。作为例子，考虑计算机 A 和 B 之间的通信，假定 B 给 A 发送一个连接请求分组，A 收到了这个分组，并发送了确认应答分组。按照两次握手的协定，A 认为连接已经成功地建立了，可以开始发送数据分组。可是，B 在 A 的应答分组在传输中被丢失的情况下，将不知道 A 是否已准备好，不知道 A 建议什么样的序列号，B 甚至怀疑 A 是否收到自己的连接请求分组。在这种情况下，B 认为连接还未建立成功，将忽略 A 发来的任何数据分组，只等待连接确认应答分组。而 A 在发出的分组超时后，重复发送同样的分组。这样就形成了死锁。
2.使用 TCP 对实时话音数据的传输有什么问题？使用 UDP 在传送数据文件时会有什么问题？
答：1.如果语音数据不是实时播放（边接受边播放）就可以使用 TCP，因为 TCP 传输可靠。接收端用 TCP 讲话音数据接受完毕后，可以在以后的任何时间进行播放。但假定是实时传输，则必须使用 UDP。
3.UDP 不保证可靠交付，但 UCP 比 TCP 的开销要小很多。因此只要应用程序接受这样的服务质量就可以使用 UDP。
4.TCP 在进行流量控制时是以分组的丢失作为产生拥塞的标志。有没有不是因拥塞而引起的分组丢失的情况？如有，请举出三种情况。
答：当 IP 数据报在传输过程中需要分片，但其中的一个数据报未能及时到达终点，而终点组装 IP 数据报已超时，因而只能丢失该数据报；IP 数据报已经到达终点，但终点的缓存没有足够的空间存放此数据报；数据报在转发过程中经过一个局域网的网桥，但网桥在转发该数据报的帧没有足够的差错空间而只好丢弃。

使用 wireshark 分析 arp 协议

一、 实验目的：

1．学习 ARP 协议的工作原理以及 ARP 分组格式；

2．学习使用 WireShark 对 ARP 协议进行分析。

练习 2：ARP 报文是直接封装在以太帧中的，为此以太帧所规定的类型字段值为___0806h_____

练习 3：对地址转换协议（ARP）描述正确的是（ D ）

A　ARP 封装在 IP 数据报的数据部分

B　发送 ARP 包需要知道对方的 MAC 地址

C　ARP 是用于 IP 地址到域名的转换

D　ARP 是采用广播方式发送的

练习 4：ARP 欺骗的原理是什么？如何进行防范？

ARP 欺骗分为二种，一种是对路由器 ARP 表的欺骗；另一种是对内网 PC 的网关欺骗。第一种 ARP 欺骗的原理是——截获网关数据。它通知路由器一系列错误的内网 MAC 地址，并按照一定的频率不断进行，使真实的地址信息无法通过更新保存在路由器中，结果路由器的所有数据只能发送给错误的 MAC 地址，造成正常 PC 无法收到信息。第二种 ARP 欺骗的原理是——伪造网关。它的原理是建立假网关，让被它欺骗的 PC 向假网关发数据，而不是通过正常的路由器途径上网。

防范措施：建立 DHCP 服务器；建立 MAC 数据库；网关机器关闭机器 ARP 动态刷新的过程，使用静态路由；网关监听网络安全。

练习 5：有人认为：“ARP 协议向网络层提供了转换地址的服务，因此 ARP 应当属于数据链路层。”这种说法为什么是错误的？

ARP 是进行地址间的转换，而 LLC 和 MAC 均没有包含这样的功能。

练习 6：ARP 协议的作用是 （A ）

A．由 IP 地址查找对应的 MAC 地址 B．由 MAC 地址查找对应的 IP 地址

C．由 IP 地址查找对应的端口号 D．由 MAC 地址查找对应的端口号

ARP 报文封装在 （A ） 中传送。

A． 以太帧 B．IP 数据报 C．UDP 报文 D．TCP 报文