

实验一：网络搭建及路由协议

计64 嵇天颖 2016010308

目录

实验一：网络搭建及路由协议

任务一：IP分配方案

任务二：网络铺设

任务三：公网网关安全

任务四：静态路由能保障网络通信

任务五：路由协议能高效保障通信

Bonus任务

Bonus Task1

Bonus Task 2

任务一：IP分配方案

修改项如下：

1. 我们为 Router2 端口2应当分配 IPv4 reserved address，而表中为它分配了公网地址。
- 因此我们要为它配置一个 IPv4 保留地址，从A类保留地址中 10.0.0.0-10.255.255.255 中选取，简单起见，为之分配 10.2.3.2。
2. 与 Router2 端口2连接的 Router3 的端口应当分配同一个子网的，我们为它分配上 10.2.3.3。
3. Server0 通过交换机 Switch0 可以联通到 Router1 的端口1上，与 PC0 , Laptop0 情况一致，因而它的 Gateway 应当是 192.168.1.1。

修改项汇总如下：

Device	Port	IP	Mask	Gateway
Router2	端口2	10.2.3.2	/24	-
Router3	端口1	10.2.3.3	/24	-
Server0	端口1	192.168.1.3	/24	192.168.1.1

任务二：网络铺设

实现过程：

1. 选择合适的设备及连线将该网络铺设出来

- 因为 Router2 需要3个端口，为了统一，我给三个路由器都加上了 HWIC-2T 模块

2. 配置 Router 路由

◦ 配置 Router1 路由

■ 配置端口1的 IP

```
1 Router>enable
2 Router#conf ter
3 Router(config)#hostname r1
4 r1(config)#interface FastEthernet 0/0
5 r1(config-if)#no shutdown
6 r1(config-if)#ip address 192.168.1.1 255.255.255.0
```

■ 配置端口2的 IP

```
1 r1(config)#interface Serial0/0/0
2 r1(config-if)#no shutdown
3 r1(config-if)#ip address 10.1.2.1 255.0.0.0
```

■ 保存一下配置

```
1 r1#copy running-config startup-config
```

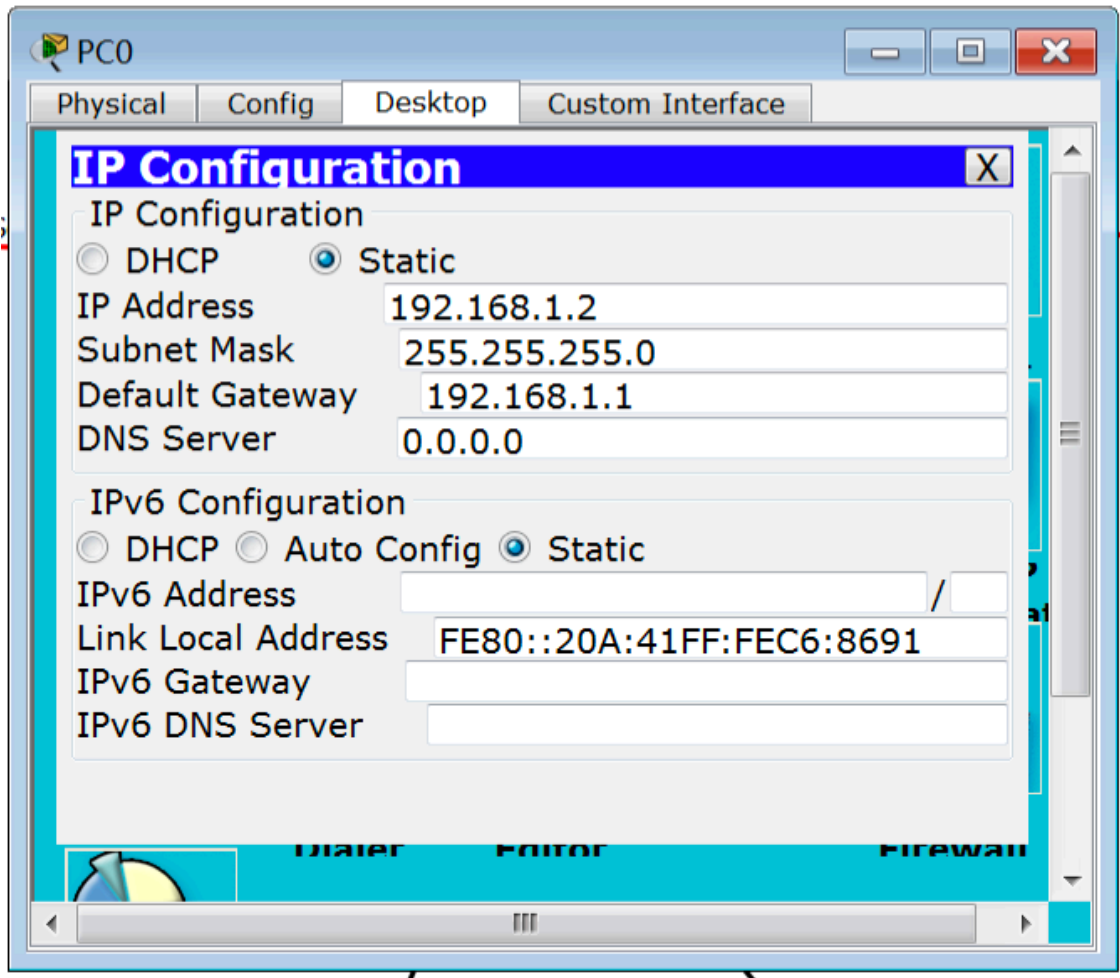
◦ 配置 Router2 和 Router3 与上面方法一致

我们可以查看配置结果， Router3 配置结果显示如下，符合我们的配置要求

```
1 r3#show ip interface brief
2 Interface          IP-Address    OK? Method Status
   Protocol
3 FastEthernet0/0    192.168.3.1   YES manual up
4 FastEthernet0/1    unassigned    YES unset  administratively down
   down
5 Serial0/0/0        10.2.3.3      YES manual up
6 Serial0/0/1        unassigned    YES unset  administratively down
   down
7 Vlan1              unassigned    YES unset  administratively down
   down
```

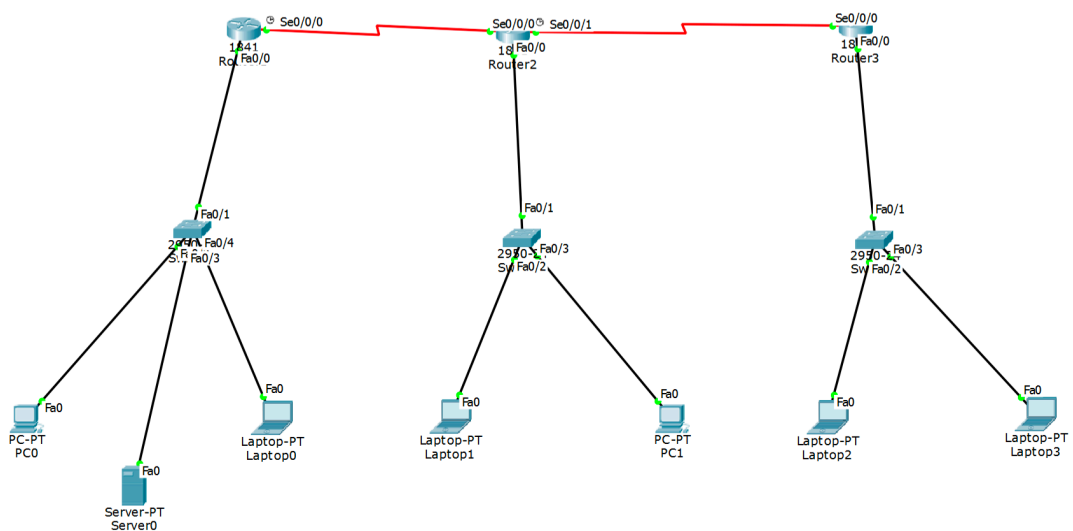
3. 配置终端 IP

- 配置 PC0，在 Desktop 的 IP Configuration 中进行配置 IP 地址，子网掩码和默认网关



- 其他的 PC, Server 和 Laptop 的配置同理

4. 最终铺设效果如图，可以看到在仿真效果下已经连通



任务三：公网网关安全

配置 Router 密码

我的清华 ID 是 jity16，所以配置的密码都是以 jity16 开头

1. 用户模式的口令 `password1:jity16@console`

通过 `console` 口进入用户模式口令设置如下：

```
1 r1(config)#line console 0
2 r1(config-line)#password jity16@console
3 r1(config-line)#login
4 r1(config-line)#end
```

2. 特权模式的口令 `password2:jity16@privilege`

通过用户模式进入特权模式的口令设置如下：

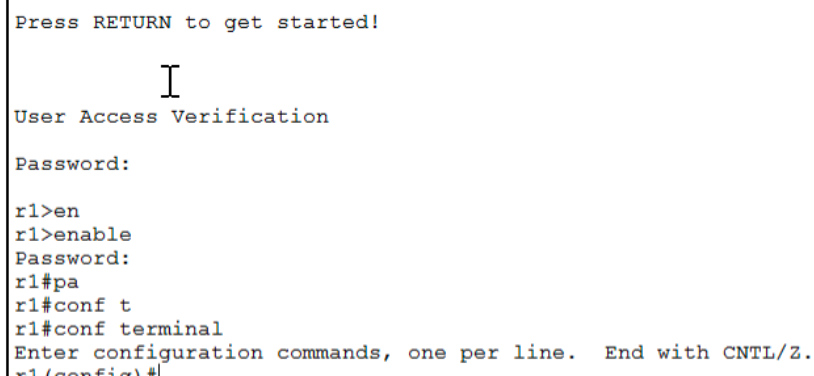
```
1 r1#en
2 r1#conf t
3 r1(config)#enable password jity16@privilege
4 r1(config)#exit
```

3. 通过 `telnet` 方式登录路由器的口令 `password3:jity16@vty`

```
1 r1(config)#line vty 0 4
2 r1(config-line)#password jity16@vty
3 r1(config-line)#login
4 r1(config-line)#end
```

4. 实际效果

尝试进行登录，如图，说明密码设置成功



```
Press RETURN to get started!

I
User Access Verification
Password:
r1>en
r1>enable
Password:
r1#pa
r1#conf t
r1#conf terminal
Enter configuration commands, one per line. End with CNTL/Z.
r1(config)#
```

如果路由器配置文件可能泄露，你的设置是否有所变化？

1. 首先我们查看我们的密码配置，发现在配置文件中密码都以明文存储：

```

1 //password 2 - privilege mode
2 enable password jity16@privilege
3
4 //password 1 - user mode
5 line con 0
6 password jity16@console
7 login
8
9 //password 3 - telnet login
10 line vty 0 4
11 password jity16@vty
12 login

```

2. 如果配置文件泄露，等于密码都被泄露了，需要对配置进行变化

- 修改一：设置特权态密码时用 `secret`

设置如下

```

1 r1(config)#enable secret jity16@secret

```

`password` 和 `secret` 同时设置时，`secret` 优先级比较高，只有 `secret` 生效，这时候配置文件变成：

```

enable secret 5 $1$mERr$9SA9d6q2jByRRQw.laPiH0
enable password jity16@privilege

```

也就是说进入特权态的密码已经被加密成功，无法从配置文件直接得到登录密码

- 修改二：这时候，我们继续观察配置文件，发现

```

line con 0
password jity16@console
login
!
line aux 0
!
line vty 0 4
password jity16@vty
login
,

```

`password1` 和 `password3` 还是明文存储，我们设法对此也进行加密处理

方法如下：

```

1 r1(config)#service password-encryption

```

此时配置文件：

- 连因为优先级没有 `secret` 高而不生效的 `password` 也被加密了

```

enable secret 5 $1$mERr$9SA9d6q2jByRRQw.laPiH0
enable password 7 082B455A10485337021905122327212F36
,

```

- 于此同时, password1 和 password3 也被加密了

```
line con 0
password 7 082B455A1048533711040217252721
login
!
line aux 0
!
line vty 0 4
password 7 082B455A10485337041F15
login
!
!
!
end
```

攻击者进行暴力破解时时间需求的变化

1. 总长六位的纯数字密码: 是1e7量级的

每位有10种可能数字, 一共有 6^{10} 种可能密码, 每种密码等概率出现, 则暴力破解时间

$$E(time) = \frac{1}{6^{10}} \times (1 + 2 + \dots + 6^{10}) = \frac{1 + 6^{10}}{2} = 30233088.5$$

2. 总长六位的混合有数字及小写字母的密码: 是1e27量级的

每位有36种可能, 一共有 6^{36} 种可能

因为要求是混合, 所以排除全部都是数字或者全部都是小写字母的情况, 也就是 $6^{36} - 6^{10} - 6^{26}$;

$$E(time) \approx 5.157 \times 10^{27}$$

3. 总长六位的混合有数字、大写字母、小写字母的密码: 是1e47量级的

每位有62种可能, 一共有 6^{62} 种可能

因为排除不是三种混合的情况对总可能数影响甚微, 所以我们不再考虑

$$E(time) \approx 8.7973 \times 10^{47}$$

4. 总长八位的混合有数字、大写字母、小写字母的密码: 是1e55量级的

每位有62种可能, 一共有 8^{62} 种可能

因为排除不是三种混合的情况对总可能数影响甚微, 所以我们不再考虑

$$E(time) \approx 4.9040 \times 10^{55}$$

任务四: 静态路由能保障网络通信

配置静态路由

1. 配置 Router1 的静态路由

因为它所不能直达的网段是 192.168.2.0 和 192.168.3.0 和 172.16.2.0

所以我们为它配置路由, 这三个网段对于 Router1 来说, 下一跳转发的 IP 都是 Router2 的端口1

```

1 r1(config)#ip route 192.168.2.0 255.255.255.0 10.1.2.2
2 r1(config)#ip route 192.168.3.0 255.255.255.0 10.1.2.2
3 r1(config)#ip route 10.2.3.0 255.255.255.0 10.1.2.2

```

配置结果查看如图：

```

r1#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

    10.0.0.0/24 is subnetted, 2 subnets
C       10.1.2.0 is directly connected, Serial0/0/0
S       10.2.3.0 [1/0] via 10.1.2.2
C       192.168.1.0/24 is directly connected, FastEthernet0/0
S       192.168.2.0/24 [1/0] via 10.1.2.2
S       192.168.3.0/24 [1/0] via 10.1.2.2
r1#

```

2. 配置 Router2 的静态路由

因为它所不能直达的网段是 192.168.1.0 和 192.168.3.0，我们为它配置路由

```

1 r2(config)#ip route 192.168.1.0 255.255.255.0 10.1.2.1
2 r2(config)#ip route 192.168.3.0 255.255.255.0 10.2.3.3

```

配置结果查看如图：

```

r2#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

    10.0.0.0/24 is subnetted, 2 subnets
C       10.1.2.0 is directly connected, Serial0/0/0
C       10.2.3.0 is directly connected, Serial0/0/1
S       192.168.1.0/24 [1/0] via 10.1.2.1
C       192.168.2.0/24 is directly connected, FastEthernet0/0
S       192.168.3.0/24 [1/0] via 10.2.3.3
r2#

```

3. 配置 Router3 的静态路由

```

1 r3(config)#ip route 192.168.1.0 255.255.255.0 10.2.3.2
2 r3(config)#ip route 192.168.2.0 255.255.255.0 10.2.3.2
3 r3(config)#ip route 10.1.2.0 255.255.255.0 10.2.3.2

```

配置结果查看如图：

```

r3#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

    10.0.0.0/24 is subnetted, 2 subnets
S       10.1.2.0 [1/0] via 10.2.3.2
C       10.2.3.0 is directly connected, Serial0/0/0
S       192.168.1.0/24 [1/0] via 10.2.3.2
S       192.168.2.0/24 [1/0] via 10.2.3.2
C       192.168.3.0/24 is directly connected, FastEthernet0/0
r3#

```

测试网络通信

1. 测试路由器间通信

- Router3 ping Router1

```

r3#ping 192.168.1.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.1, timeout is 2 seconds:
!!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 10/15/26 ms

r3#

```

- Router1 ping Router3

```

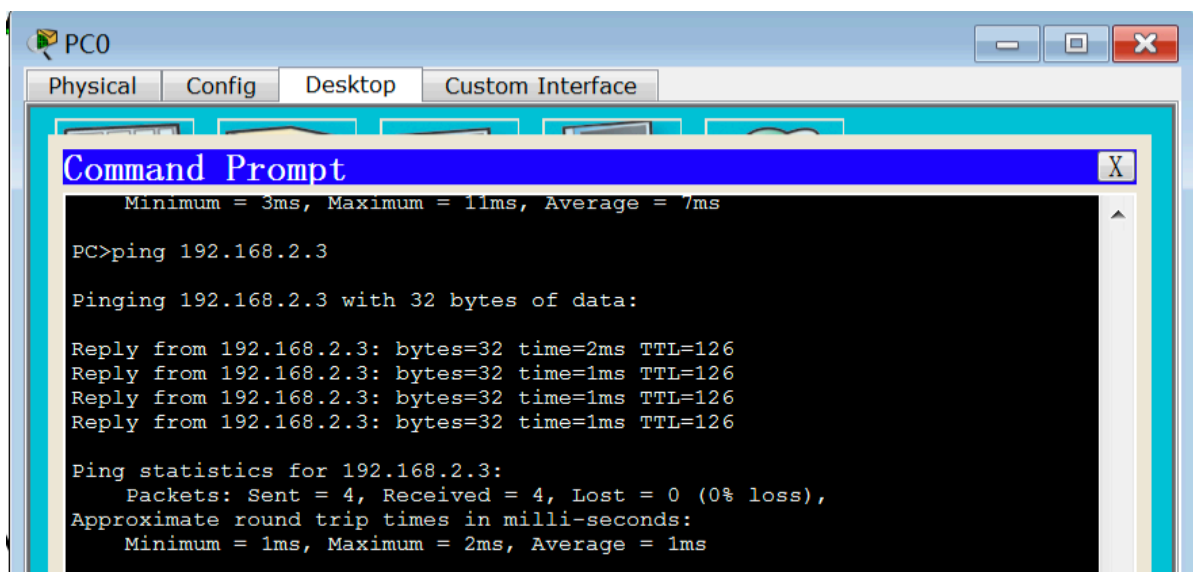
r1#ping 192.168.3.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.3.1, timeout is 2 seconds:
!!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/8/14 ms

r1#

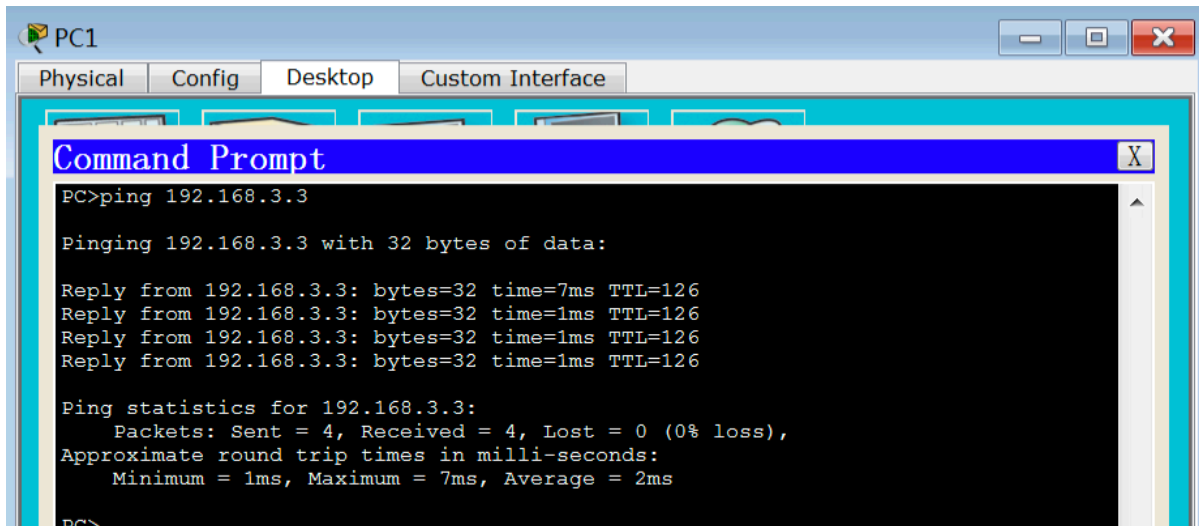
```

2. 测试不同网段的终端通信

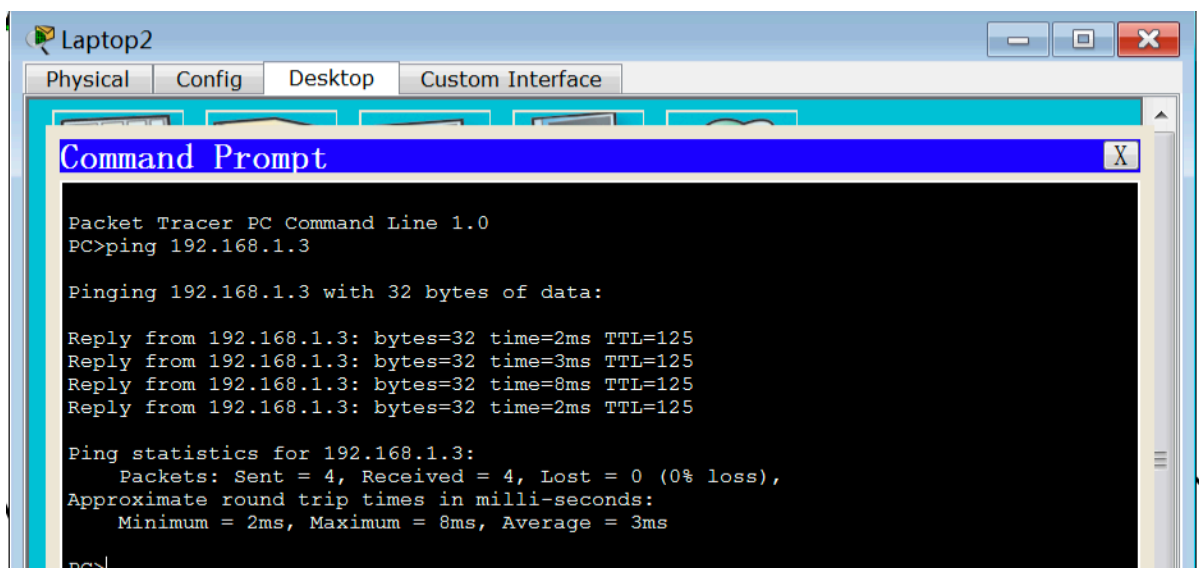
- PC0 ping Laptop1



- PC1 ping Laptop 3



- o Laptop2 ping Server0



结论

根据上面的测试，我们可以发现静态路由已经配置好，并且没有冗余的静态路由，不同部门间可以实现网络间通信。

任务五：路由协议能高效保障通信

问题解答：

RIP 协议限制网络跳数在16跳以内，公司里终端设备加起来不到16台，也就是最坏情况下网络直径最大为15，所以 RIP 协议能够满足要求

配置 RIP 协议

公司目前的局域网用 RIP 协议可以满足要求，最大跳数远小于16跳。

1. 配置 Router1

- 首先删除配置好的静态路由：

```
1 r1(config)#no ip route 192.168.2.0 255.255.255.0 10.1.2.2
2 r1(config)#no ip route 192.168.3.0 255.255.255.0 10.1.2.2
3 r1(config)#no ip route 10.2.3.0 255.255.255.0 10.1.2.2
```

- 进入 RIP 配置模式

```
1 r1(config)#router rip
```

- 宣告直连网段

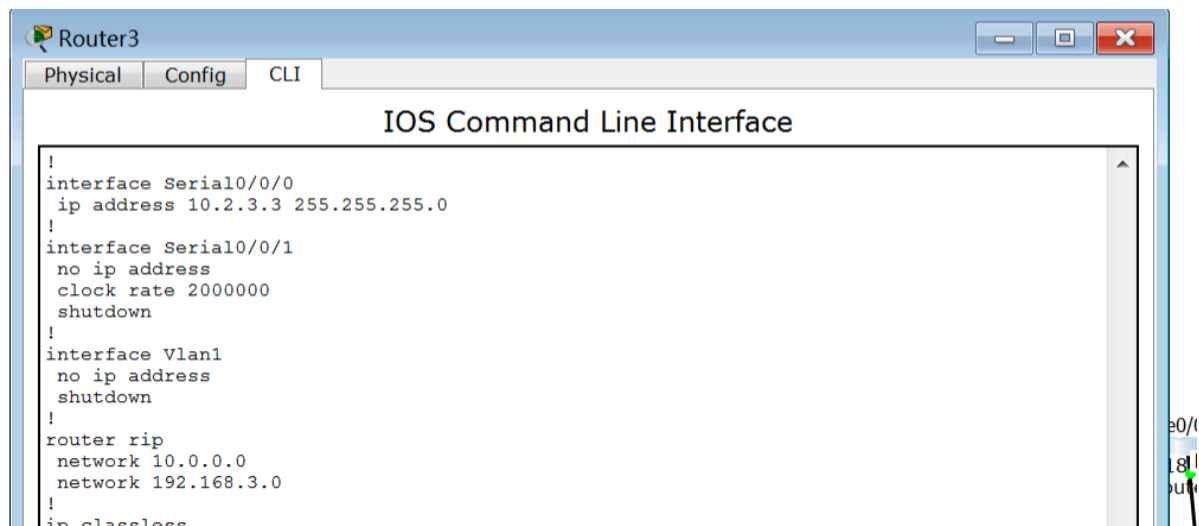
由于 RIP 不支持设置变长子网掩码，所以 10.x.x.x 开头的地址都会被强制设定为 10.0.0.0，所以在配置时直接配置称 10.0.0.0 即可

```
1 r1(config)#router rip
2 r1(config-router)#network 192.168.1.0
3 r1(config-router)#network 10.0.0.0
```

2. 配置 Router2 和 Router3 的方法与上面雷同

配置结果

- 我们先查看配置文件（这里展示 Router3 的配置文件，可以看到宣告的直连网段）



- 我们查看 RIP 路由表，可以看到 Router3 路由表已成功学习

```
r3#show ip route rip
 10.0.0.0/24 is subnetted, 2 subnets
R   10.1.2.0 [120/1] via 10.2.3.2, 00:00:06, Serial0/0/0
R   192.168.1.0/24 [120/2] via 10.2.3.2, 00:00:06, Serial0/0/0
R   192.168.2.0/24 [120/1] via 10.2.3.2, 00:00:06, Serial0/0/0
r3#
```

- 进行网络通信测试 Router3 ping Router1, 网络通信正常

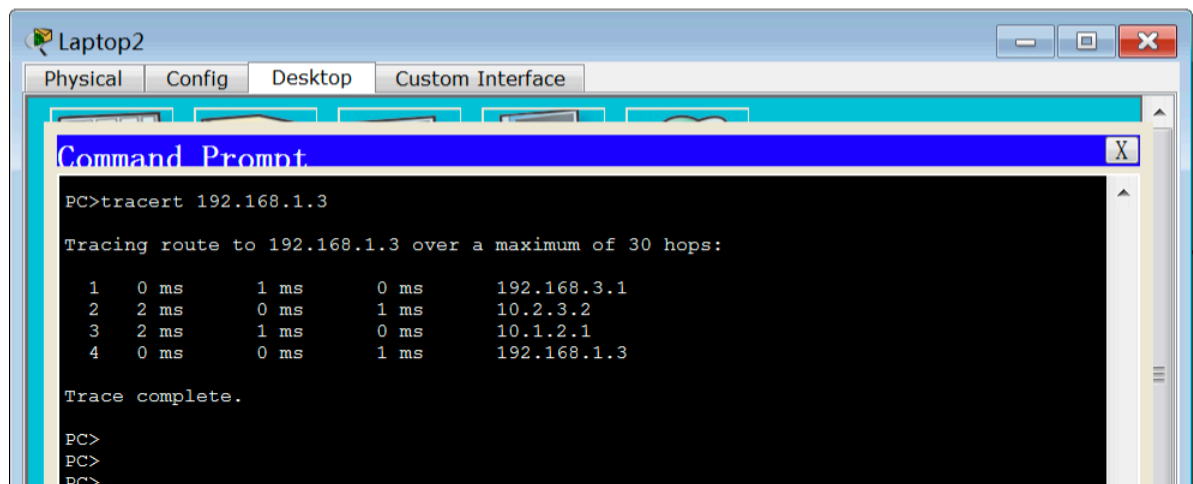
```
r3#ping 192.168.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.1, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 15/20/34 ms

r3#
```

- 用 `tracert` 进行路径追踪(从 192.168.3.2 到 192.168.1.3)

可以发现经过四跳到达目的地址



Bonus任务

Bonus Task1

起先我所设置的 `secret` 密码是 `jity16@secret`, 查看配置文件:

```
enable secret 5 $1$mERr$9SA9d6q2jByRRQw.laPiH0
enable password 7 082B455A10485337021905122327212F36
```

发现采用的是 `secret 5`, 查阅发现采用的是 `MD5` 算法

我在 `MAC OS` 下用 `openssl` 进行了加密验证

```
jitianying — -bash — 102x24
(base) jitianyingdeMacBook-Pro:~ jitianying$ openssl passwd -1 -salt mERr -table jity16@secret
jity16@secret $1$mERr$9SA9d6q2jByRRQw.laPiH0
(base) jitianyingdeMacBook-Pro:~ jitianying$
```

发现加密得到的结果一致

Bonus Task 2

结论: `ARP` 广播过程会造成丢包, 在实验环境下, 可能会造成第一次无法 `ping` 通的现象

分析：在实验时，我第一次用 ping 的时候出现了不通的现象，是 PC0 ping PC1 的时候，我就用这个过程为例分析

1. 起初 PC0 只知道目的 IP，不知道 Router 0 的 MAC 地址，发送 ARP 广播请求获取 Router 0 的 MAC 地址
2. Router 0 的 F0/0 端口接收到 ARP 广播，查看数据帧发现是自己的 IP 地址，F0/0 端口作出回应，PC0 收到回应并进行 ARP 缓存
3. PC0 发送带有目的 MAC 地址的 ping 包，Router 0 的 F0/0 端口接收到并发现目的 MAC 地址是自己，然后开始查看路由表进行匹配。而我们已经配置好了静态路由，所以它知道要发送的方向。
4. 此时需要重新封装 MAC 头部信息，但是不知道目标 MAC 地址，只能丢弃 ping 包，并在端口 Se 0/0/0 发起 ARP 广播
5. Router2 接收到广播，发现目标 IP 是自己，给出回应也就是自己的 MAC 地址。但它也不知道 PC1 的 MAC 地址，所以只能丢弃 ping 包，并在端口 F0/0 发起 ARP 广播，获取了 PC1 的 MAC 地址
6. 此时 PC1 收到该广播，拆包发现目标 IP 是自己的，对 Router1 的 F0/0 端口给出回应，告诉它自己的 MAC 的地址，此时 Router1 的 F0/0 端口缓存该回应的信息
7. 此时 PC0 的 ping 包数据再次过来，终于不会被丢弃了，成功了

所以，这个广播过程中会丢弃 ping 包，所以很有可能在第一次 ping 的时候失败，后来又能成功是因为缓存的 ARP 表里面已经记录了目的 MAC 地址，可以不丢包地发送了。