

# Task

## A

---

### Find & Fix

#### error1

R1's route table is like:

```
default via 163.5.5.1 dev eth3
55.135.72.0/24 via 220.129.51.1 dev eth0
67.124.83.0/24 dev eth2 proto kernel scope link src 67.124.83.1
163.5.5.0/24 dev eth3 proto kernel scope link src 163.5.5.2
164.220.97.0/29 via 189.81.86.10 dev eth1
189.81.86.0/24 dev eth1 proto kernel scope link src 189.81.86.4
220.129.51.0/24 dev eth0 proto kernel scope link src 220.129.51.2
```

No rule for subnet `164.220.97.0/24`, and the 5th row is similar, so it might be a mistake. We have 2 options to fix this:

1. add a new rule:

```
ip route add 164.220.97.0/24 via 189.81.86.10
```

2. change the 5th row like this:

```
164.220.97.0/24 via 189.81.86.10 dev eth1
```

Now the package can pass from 55.135.72.0/24 to 164.220.97.0, but `ping` still failed, so let's check R2.

#### error2

R2's route table says `55.135.72.0/24 via 67.124.83.1 dev eth2`, which means package will go to R4, however R4's route table is like:

```
default via 163.5.5.1 dev eth3
67.124.83.0/24 dev eth2 proto kernel scope link src 67.124.83.1
163.5.5.0/24 dev eth3 proto kernel scope link src 163.5.5.2
164.220.97.0/24 via 189.81.86.10 dev eth1
189.81.86.0/24 dev eth1 proto kernel scope link src 189.81.86.4
220.129.51.0/24 dev eth0 proto kernel scope link src 220.129.51.2
```

It doesn't know how to reach subnet 55.135.72.0/24. So we have 2 options to fix this problem:

1. change R2's route table to :

```
55.135.72.0/24 via 189.81.86.1
```

2. add a rule in R4:

```
ip route add 55.135.72.0/24 via 220.129.51.1
```

Now `ping 189.81.86.10` at R1 can be successful.

### error3

But ping webserver still fail, so let's check webserver's configure. To webserver, it's subnet mask is wrong:

```
164.220.97.12/30
fix:
164.220.97.12/24
```

### error4

To webserver2, webserver3 and sshserver, their ip route like:

```
default via 164.220.97.1 dev eth0
```

However, the ip address of interface of R2 is `164.220.97.4/24`, and there is no `164.220.97.1/24`, so we have 2 options to fix this problem:

1. change R2 eth1's ip address:

```
164.220.97.1/24
```

2. change `default` :

```
ip route delete default
ip route add default via 164.220.97.4
```

## error5

The last error is that dnsserver does not connect to R2, because they are in different subnet. We can fix this problem by giving dnsserver a new ip address in subnet 92.170.73.0/24, according to the `maradns` configuration, the new ip address should be:

```
92.170.73.100/24
```

## Test

1. Ping webserver, sshserver, dnsserver at client, success;
2. Test webserver service by using `lynx 164.220.97.12`, success;

Since there are no http service running on webserver2 and webserver3, `lynx` to them will fail. But this is not a error.

## B

---

- Add route rule in **R1** and **R2**

```
# R1
ip route add default via 220.129.51.2

# R2
ip route add default via 67.124.83.1
```

- Add route rule in **R3**

```
# to each subnet
ip route add 55.135.72.0/24 via 163.5.5.2
ip route add 164.220.97.0/24 via 163.5.5.2
ip route add 92.170.73.0/24 via 163.5.5.2
```

- Add route rule in **R4**

```
# let R4 know the subnet 92.170.73.0/24
ip route add 92.170.73.0/24 via 67.124.83.2
```

- Test
  1. add new client **n3** in outer network;
  2. from **n3** ping every subnet;

## C

---

Edit firewall rules on **R4**

- Add default rule to drop all packages

```
iptables -P FORWARD DROP
```

- Enable other interface except eth1 (subnet 164.220.97.0/24)

```
iptables -A FORWARD ! -o eth1 ! -i eth1 -j ACCEPT
```

- Allow ssh (tcp port 22)

```
iptables -A FORWARD -p tcp --sport 22 -j ACCEPT  
iptables -A FORWARD -p tcp --dport 22 -j ACCEPT
```

- Allow http (tcp port 80)

```
iptables -A FORWARD -p tcp --sport 80 -j ACCEPT  
iptables -A FORWARD -p tcp --dport 80 -j ACCEPT
```

## Test

- Let client ping dnsserver, success.
- Let client ping webserver, faild. ()

## D

---

See in assignment2-error-x27505928.imn.