

# 南京大学本科生实验报告

课程名称：计算机网络

任课教师：田臣/李文中

助教：

学院	计算机科学与技术系	专业（方向）	计算机科学与技术
学号	211220079	姓名	谷石磊
Email	211220079@smail.nju.edu.cn	开始/完成日期	2023.4.15

## 1. 实验名称

Lab 3: Respond to ARP

## 2. 实验目的

修改 myrouter.py 文件，实现对 ARP 包的处理，并建立 arp 表。

## 3. 实验内容

Task 2: Handle ARP Request

收到一个包后，判断该包是否为 arp 的 request，并做出相应的回应。

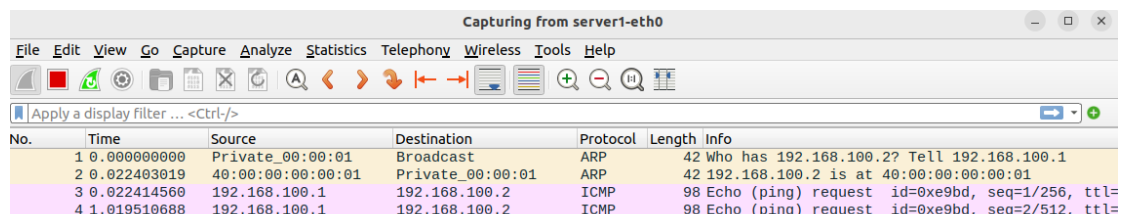
Task 3: Cached ARP Table

建立一个 arp 缓存表，记录 IP 和 MAC 地址的对应关系。

## 4. 实验结果

Task 2: Handle ARP Request

在 mininet 中令 server1 使用命令 `ping -c2 192.168.100.2`，使用 wireshark 捕获的 server1 结果如下：



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	Private_00:00:01	Broadcast	ARP	42	Who has 192.168.100.2? Tell 192.168.100.1
2	0.022403019	40:00:00:00:00:01	Private_00:00:01	ARP	42	192.168.100.2 is at 40:00:00:00:00:01
3	0.022414560	192.168.100.1	192.168.100.2	ICMP	98	Echo (ping) request id=0xe9bd, seq=1/256, ttl=
4	1.019510688	192.168.100.1	192.168.100.2	ICMP	98	Echo (ping) request id=0xe9bd, seq=2/512, ttl=

可以发现有两个 ARP 包，分别是 server1 发送的 request 包，用来询问 192.168.100.2 对应的 MAC 地址，和 router 发送的 reply，回应该 request。下面两个 ICMP 为发送的两次 ping。从结果可知，router 正确地处理了发送来的 ARP 的 request 包，做了正确的回应。

测试结果如下：

```
Passed:
1  ARP request for 192.168.1.1 should arrive on router-eth0
2  Router should send ARP response for 192.168.1.1 on router-eth0
3  An ICMP echo request for 10.10.12.34 should arrive on router-eth0, but it should be dropped (router should only handle ARP requests at this point)
4  ARP request for 10.10.1.2 should arrive on router-eth1, but the router should not respond.
5  ARP request for 10.10.0.1 should arrive on on router-eth1
6  Router should send ARP response for 10.10.0.1 on router-eth1

All tests passed!
```

### Task 3: Cached ARP Table

我们使用 ping 让 client、server1 和 server2 互相 ping 一次，截图如下：

```
er-eth1
23:21:33 2023/04/18 INFO ARP table
23:21:33 2023/04/18 INFO 10.1.1.1:30:00:00:00:00:01
23:22:25 2023/04/18 INFO ARP table
23:22:25 2023/04/18 INFO 10.1.1.1:30:00:00:00:00:01
23:22:25 2023/04/18 INFO 192.168.200.1:20:00:00:00:00:01
23:22:41 2023/04/18 INFO ARP table
23:22:41 2023/04/18 INFO 10.1.1.1:30:00:00:00:00:01
23:22:41 2023/04/18 INFO 192.168.200.1:20:00:00:00:00:01
23:22:41 2023/04/18 INFO 192.168.100.1:10:00:00:00:00:01
```

可以发现，client ping 一次后，router 的 arp 表学习了 client 的 IP 地址对应的 MAC 地址；之后 server2 ping 一次，router 的 arp 表学习了 server2 的 IP 地址对应的 MAC 地址；之后 server1 ping 一次，router

的 arp 表学习了 server2 的 IP 地址对应的 MAC 地址；

## 5. 核心代码

### Task 2: Handle ARP Request

```
arp = packet.get_header(Arp)
if arp:
    # debugger()
    self.table.add(arp.senderprotoaddr, arp.senderhwaddr)
    if arp.operation == ArpOperation.Request:
        ip = arp.targetprotoaddr
        try:
            port = self.net.port_by_ipaddr(ip)
            pkt = create_ip_arp_reply(port.ethaddr,
arp.senderhwaddr,
                                port.ipaddr,
arp.senderprotoaddr)
            self.net.send_packet(ifaceName, pkt)
        except KeyError as e:
            log_info(e)
```

### Task 3: Cached ARP Table

使用一个字典来存，key 为 IP 地址，value 为 MAC 地址。

```
class ARPtable():

    def __init__(self):
        self.table = {}

    def add(self, ip, mac):
        if mac != 'ff:ff:ff:ff:ff:ff':
            self.table[ip.exploded] = mac.toStr()
            self.print_table()

    def print_table(self):
        log_info('ARP table')
        for k, v in self.table.items():
            log_info(f'{k}:{v}')

    def query(self, ip):
        return self.table.get(ip.exploded)
```

## 6. 总结与感想

本次实验对我来说有一定的挑战性，我也花了一些时间才基本完成。总的来说，收获很大，不仅对 `router` 有了更深的理解，也对 `python` 的类和字典有了更多的了解。