

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	计算机科学与技术
学号	185220001	姓名	磯田智明
Email	185220001@smail.nju.edu.cn	开始/完成日期	2021.04.07/2021.04.09

1. 实验名称

IPv4 Router-Respond to ARP

2. 实验目的

- 对分配给路由器接口地址的ARP请求作出响应。

3. 实验内容

Task 1: Preparation

配置实验环境

Task 2: Handle ARP Request

处理接收到的ARP协议

Task 3: Cached ARP Table

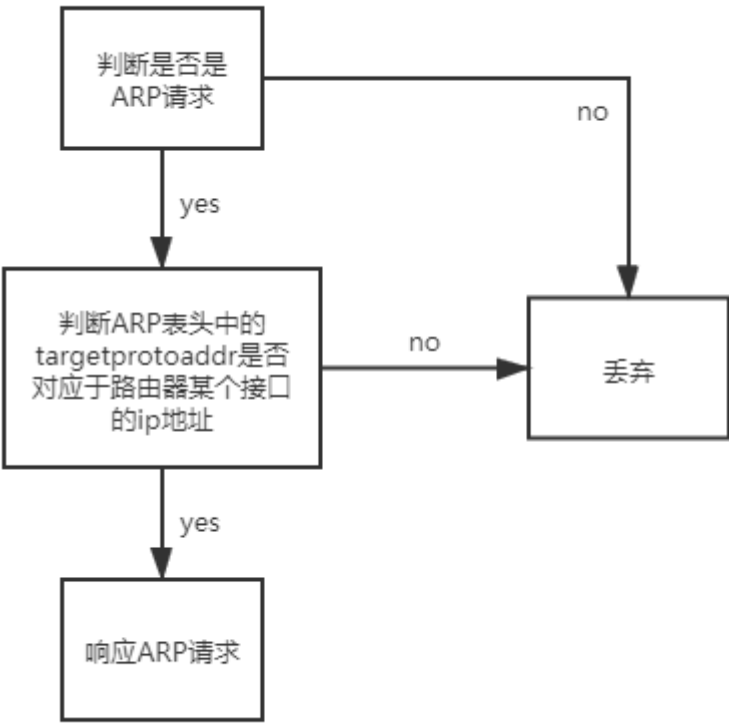
创建一个记录IP和MAC地址的ARP表

4. 实验结果

Task 2: Handle ARP Request

logic of responding to the ARP request

首先要判断传来的包是否为ARP请求；如果传来的包是ARP请求并且表头中的目的ip地址是路由器中的接口；如果以上条件均满足就调用 `create_ip_arp_reply` 函数响应ARP请求



Running in the Test Environment

```
(syenv) njucs@njucs-VirtualBox:~/sy/lab-3-vectormoon$ swyard -t testcases/myrouter1 testsenario.srpy myrouter.py
00:55:10 2021/04/08 INFO Starting test scenario testcases/myrouter1 testsenario.srpy
00:55:10 2021/04/08 INFO -----10:00:00:00:00:01-----
00:55:10 2021/04/08 INFO -----192.168.1.1-----
00:55:10 2021/04/08 INFO Sending packet Ethernet 10:00:00:00:00:01->30:00:00:00:00:01 ARP | Arp 10:00:00:00:00:01:192.168.1.1 30:00:00:00:00:01:192.168.1.100 to router-eth0
00:55:10 2021/04/08 INFO -----10:00:00:00:00:01-----
00:55:10 2021/04/08 INFO No ARP packet
00:55:10 2021/04/08 INFO -----10:00:00:00:00:02-----
00:55:10 2021/04/08 INFO -----172.16.42.1-----
00:55:10 2021/04/08 INFO -----10:00:00:00:00:02-----
00:55:10 2021/04/08 INFO -----10.10.0.1-----
00:55:10 2021/04/08 INFO Sending packet Ethernet 10:00:00:00:00:02->70:00:ca:fe:c0:de ARP | Arp 10:00:00:00:00:02:10.10.0.1 70:00:ca:fe:c0:de:10.10.5.5 to router-eth1

Results for test scenario ARP request: 6 passed, 0 failed, 0 pending

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!
```

Running in the Mininet

在终端中输入以下指令启动mininet

```
$ sudo python start_mininet.py
```

在mininet中启动 `server1` 和 `router`

```
mininet> xterm server1
mininet> xterm client
```

用wireshark抓 `server1` 的包

```
server1# wireshark -k &
```

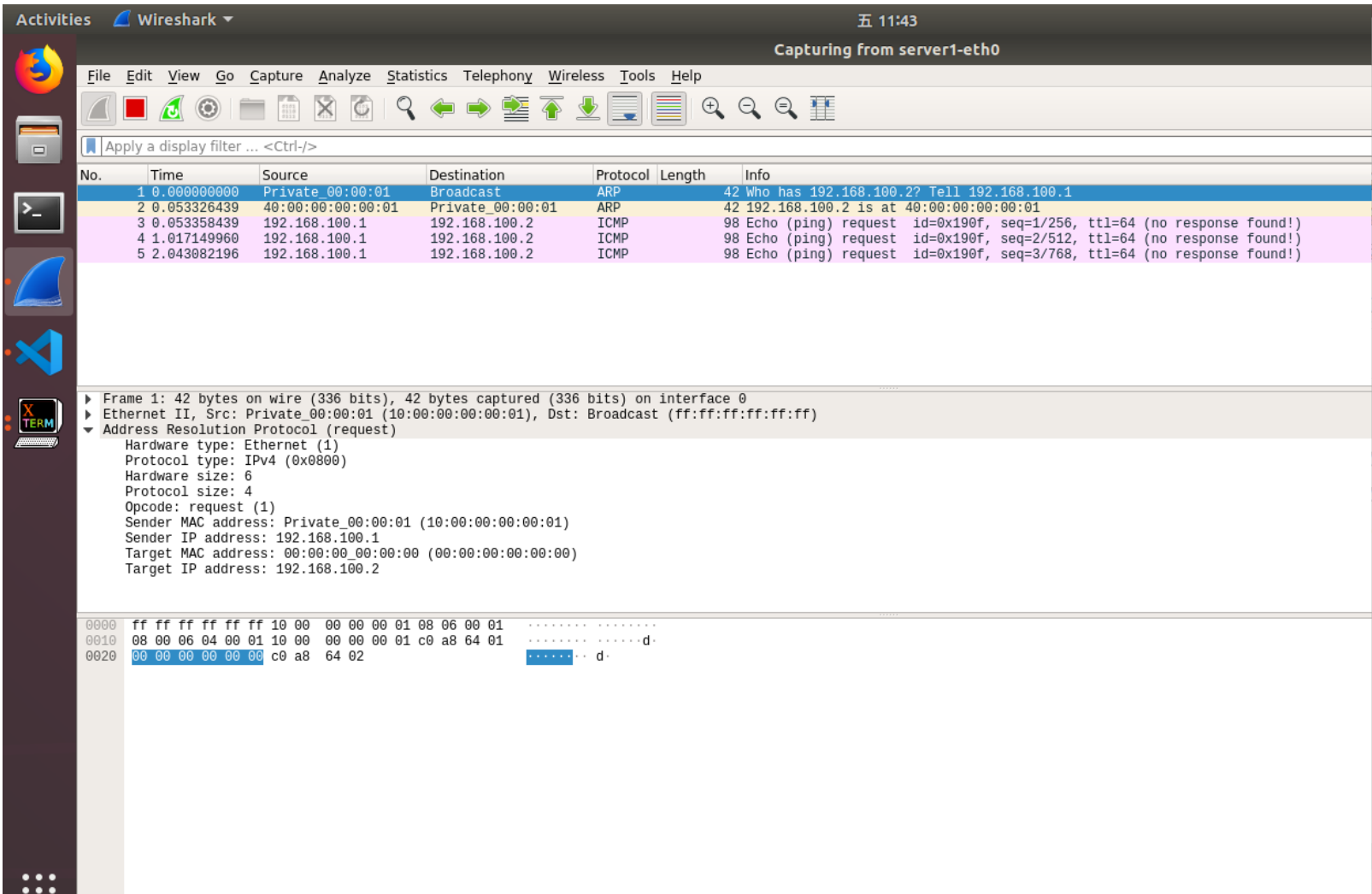
在 `router` 的xterm中打开虚拟环境并且启动 `my_router.py`

```
(syenv) router# swyard myrouter.py
```

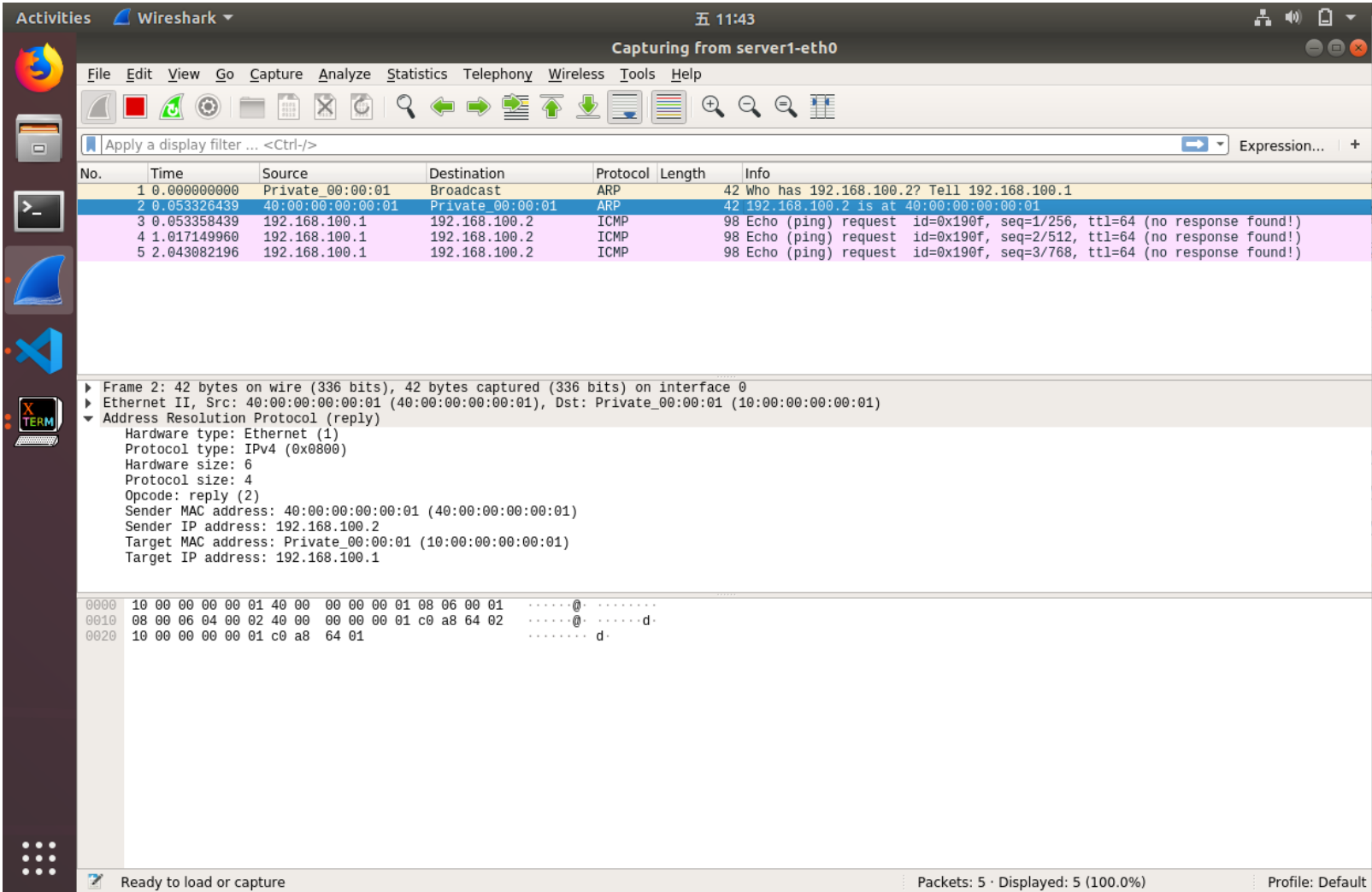
在 `server1` 的xterm中，使用以下指令测试

```
server1# ping -c3 192.168.100.2
```

得到下图结果：



在Time为0时，server1要向router发送数据，但是server1并不知道router的mac地址，所以使用ARP协议广播，目的是找到router的mac地址。具体情况如上图所示，此时的 sender MAC address 和 Sender IP address 都是server1的信息； Target MAC address 为0是因为目前还不知道router的mac地址是什么，所以暂时填零。 Target IP address 即为要发送的目的ip地址



在Time为5时，router要向server1回应ARP协议所以发送数据包。具体情况如上图所示，此时的 sender MAC address 和 Sender IP address 都变更为router的信息； Target MAC address 和 Target IP address 都变更为server1的信息

Task 3: Cached ARP Table

使用dict保存arp表

使用如下指令测试arp表：

```
mininet> client ping -c1 router
mininet> server1 ping -c1 router
mininet> server2 ping -c1 router
```

得到如下输出：

```
16:05:38 2021/04/09 INFO Saving iptables state and installing switchyard rules
16:05:38 2021/04/09 INFO Using network devices: router-eth1 router-eth0 router-eth2
16:05:48 2021/04/09 INFO -----arp_table_info-----
16:05:48 2021/04/09 INFO IP: 10.1.1.1; MAC Address: 30:00:00:00:00:01
16:05:48 2021/04/09 INFO -----
16:05:48 2021/04/09 INFO Sending packet Ethernet 40:00:00:00:00:03->30:00:00:00:00:01 ARP | Arp 40:00:00:00:00:03:10.1.1.2 30:00:00:00:00:01:10.1.1.1 to router-eth2
16:05:48 2021/04/09 INFO No ARP packet
16:07:46 2021/04/09 INFO -----arp_table_info-----
16:07:46 2021/04/09 INFO IP: 10.1.1.1; MAC Address: 30:00:00:00:00:01
16:07:46 2021/04/09 INFO IP: 192.168.100.1; MAC Address: 10:00:00:00:00:01
16:07:46 2021/04/09 INFO -----
16:07:46 2021/04/09 INFO Sending packet Ethernet 40:00:00:00:00:01->10:00:00:00:00:01 ARP | Arp 40:00:00:00:00:01:192.168.100.2 10:00:00:00:00:01:192.168.100.1 to router-eth0
16:07:46 2021/04/09 INFO No ARP packet
16:08:03 2021/04/09 INFO -----arp_table_info-----
16:08:03 2021/04/09 INFO IP: 10.1.1.1; MAC Address: 30:00:00:00:00:01
16:08:03 2021/04/09 INFO IP: 192.168.100.1; MAC Address: 10:00:00:00:00:01
16:08:03 2021/04/09 INFO IP: 192.168.200.1; MAC Address: 20:00:00:00:00:01
16:08:03 2021/04/09 INFO -----
16:08:03 2021/04/09 INFO Sending packet Ethernet 40:00:00:00:00:02->20:00:00:00:00:01 ARP | Arp 40:00:00:00:00:02:192.168.200.2 20:00:00:00:00:01:192.168.200.1 to router-eth1
16:08:04 2021/04/09 INFO No ARP packet
```

在一条新的arp包传入，就检查并更新对应ip地址和mac地址的值

5. 核心代码

Task 2: Handle ARP Request

```
targetip_exist_flag = -1
for i in range(len(self.ip_list)):
    if (self.ip_list[i] == arp.targetprotoaddr):
        targetip_exist_flag = i

    if (targetip_exist_flag != -1):
        arp_reply_pkt = create_ip_arp_reply(input_port.ethaddr,
        arp.senderhwaddr, arp.targetprotoaddr, arp.senderprotoaddr)
        self.net.send_packet(ifaceName, arp_reply_pkt)
        log_info (f"Sending packet {arp_reply_pkt} to {ifaceName}")
```

首先检查传来的arp包目的地址是否为路由器对应的端口，如果是，则使用 `create_ip_arp_reply` 响应ARP请求

Task 3: Cached ARP Table

```
self.arp_table[arp.senderprotoaddr] = arp.senderhwaddr
```

在一个新的键值出现的时候就会增加一对键值，如果是已有键值但是对应的端口信息不同则会更新

6. 总结与感想

在做实验的过程中，明显的感觉出路由器和交换机、集线器的不同，路由器给人的感觉更像是一个与其他节点相似但是具有转发功能的节点；并且发送的包的地址都是指向路由器的，并非指向其他节点