

计网实验三

实验名称：

Respond to ARP

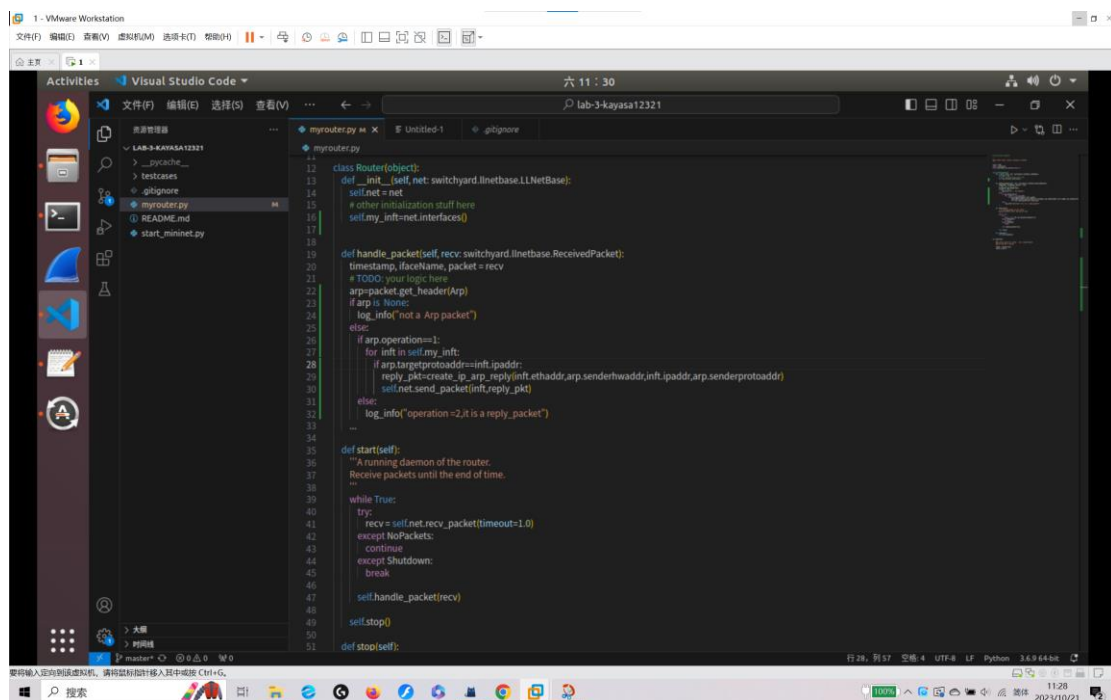
实验目的：

了解路由器对于 arp 的相应与缓存

实验具体步骤：

TASK TWO

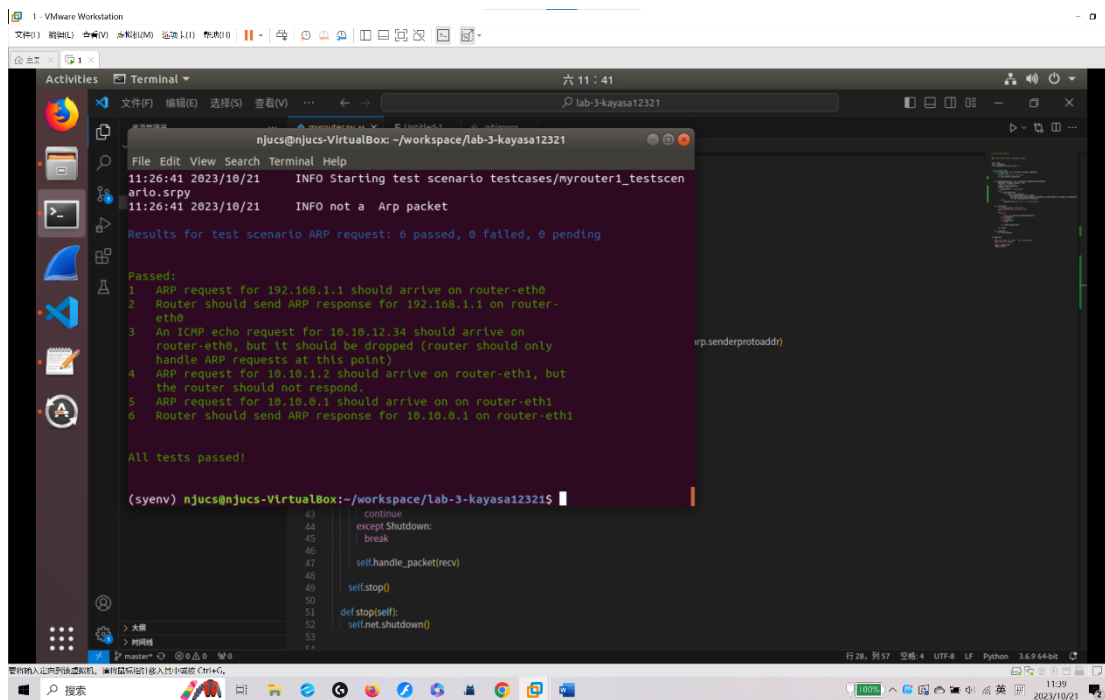
步骤一：实现 Arp 请求的逻辑



根据手册要求,我们首先判断他是不是一个 arp 的数据包,如果不是,在此次的实验要求中,我们规定需要忽略处理。

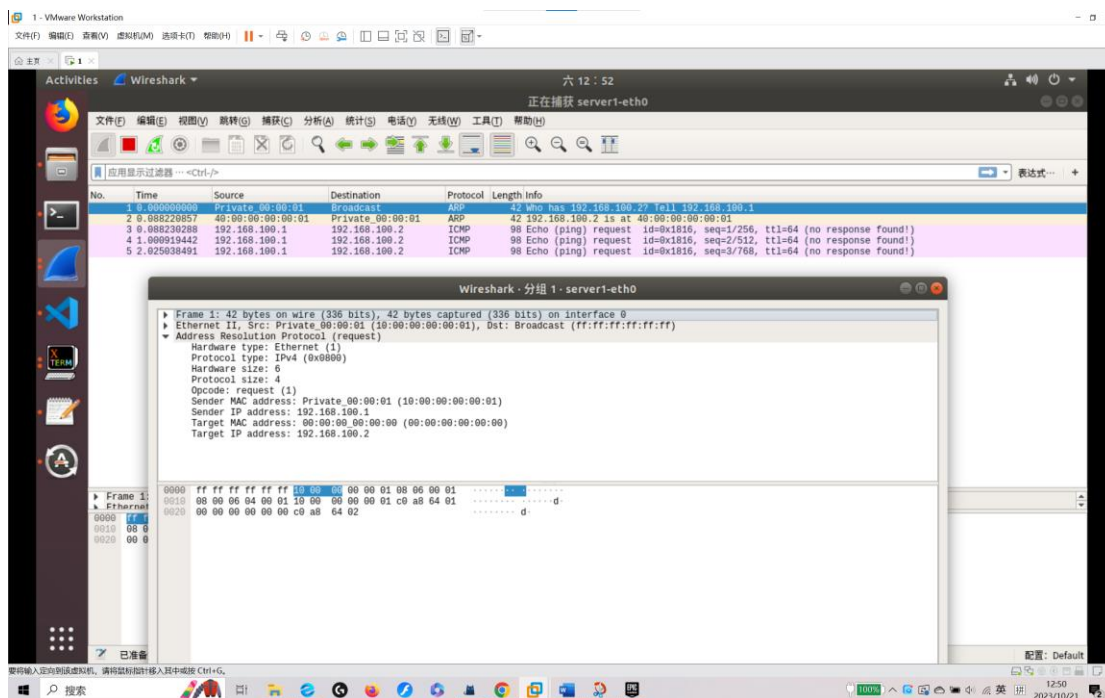
然后在判断出其是 ARP 请求,我们需要判断它是 replay 还是 request,如果是 replay 型的话目的地址应该不是分配给路由器的所以我们直接忽略,然后如果他是 request 的类型,我们需要将目的 ip 地址和所有的路由器接口匹配,看哪个的 interface 的 mac 地址与 arp 中的 targetprotoaddr 相匹配,如果判断出相匹配,直接采用手册中提供的 create_ip_arp_reply 来构造回复数据包,之后直接通过 send_packet 函数来发送。

步骤二：

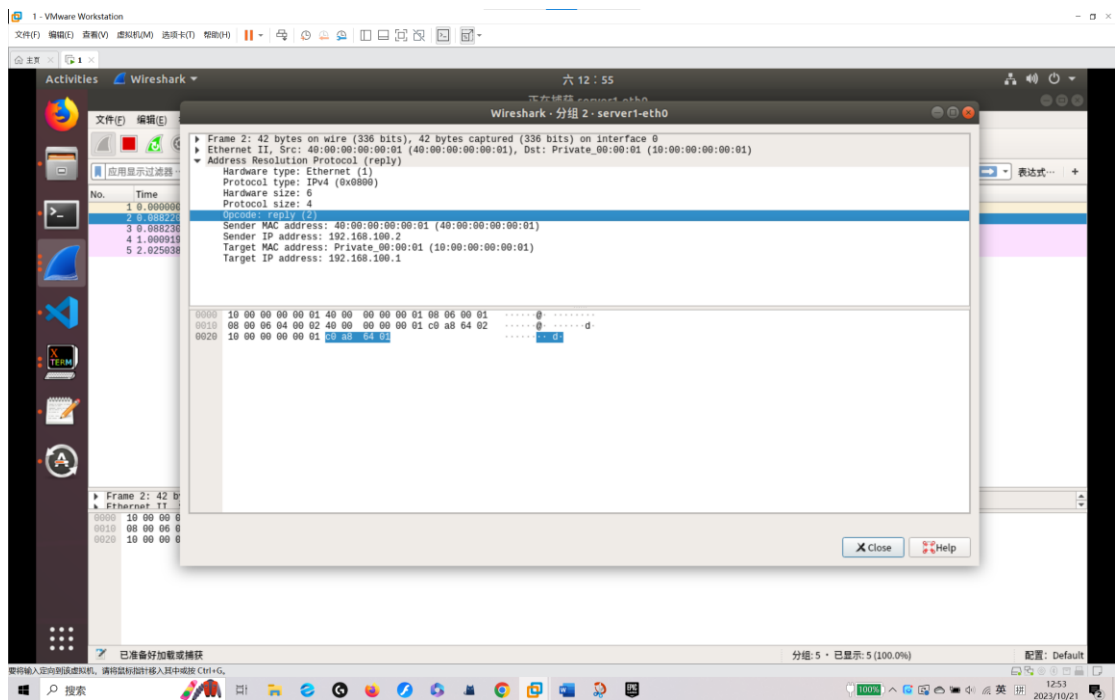


步骤三:

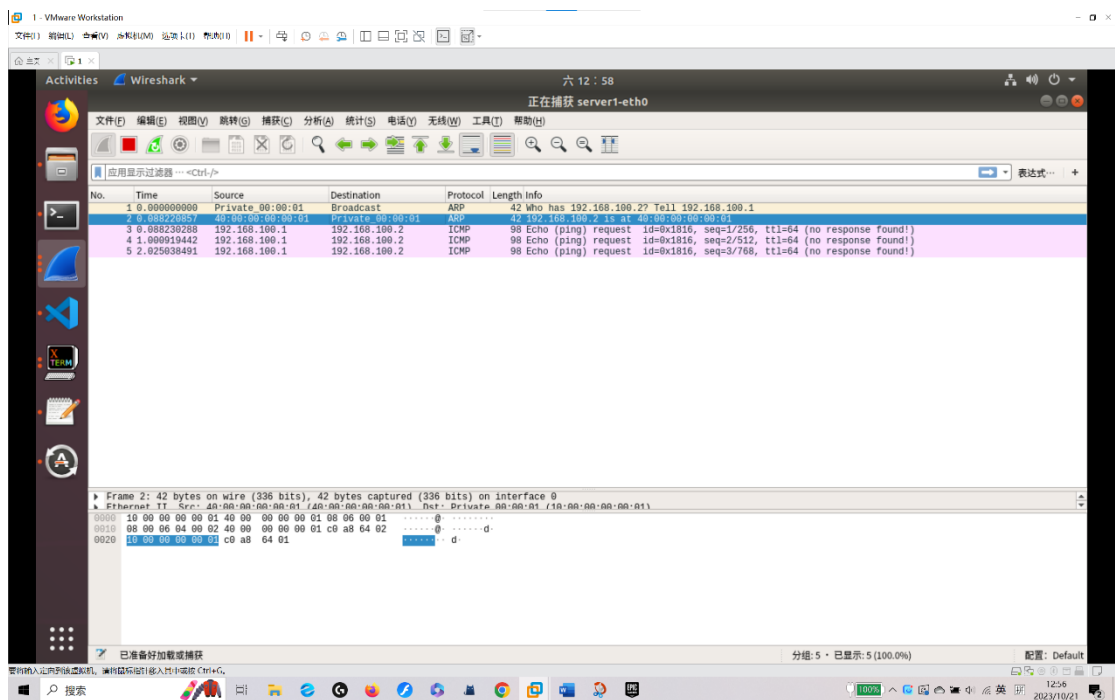
从 server1 ping c3 192.168.100.2



首先按照手册上的拓扑就是在本 lab3 所构建的拓扑，首先发送 arp 请求，即相当于发送 request 数据包，此时目标 mac 地址是全零，因为这时候他在请求目的 IP 地址



此时路由器接口匹配到目标的 IP 地址，此时路由器发送 arp 的 reply，从该对应的接口发送回一开始发出 arp 的请求的数据包。



```
File "Node: router"
('r root@njucs-VirtualBox:~/workspace/lab-3-kayasa12321# cd
*** root@njucs-VirtualBox:~# cd workspace
*** root@njucs-VirtualBox:~/workspace# source ./syenv/bin/activate
(syenv) root@njucs-VirtualBox:~/workspace# source ./syenv/bin/activate
(syenv) root@njucs-VirtualBox:~/workspace# cd lab-3-kayasa12321
*** (syenv) root@njucs-VirtualBox:~/workspace/lab-3-kayasa12321# swyard myrouter.py
12:26:47 2023/10/21 INFO Saving iptables state and installing switchyard rules
*** 12:26:47 2023/10/21 INFO Using network devices: router-eth2 router-eth0 router-eth1
*** 12:27:25 2023/10/21 INFO not a Arp packet
*** 12:27:26 2023/10/21 INFO not a Arp packet
net 12:27:27 2023/10/21 INFO not a Arp packet
***
net
***
net
***
net
***
net
***
net
***
server1 : ('sysctl -w net.ipv6.conf.default.disable_ipv6=1')
```

此时在对应好之后，进行了三次的测试链接。

TASK THREE:

采用一个一键多值的字典，key 为 sender 的 ip 地址，其对应的值为 sender 的 mac 地址以及其被加入缓存表的时间，每次都进行判断，用当前的时间对 table 内的每一项的加入时间比较，如果大于 10 则被删除。然后判断其为 arp 请求后，将该 sender 的记录进去，如果是已经在内就是直接更新。

```

19
20 def handle_packet(self, rcv: switchyard.llnetbase.ReceivedPacket):
21     timestamp, ifaceName, packet = rcv
22     # TODO: your logic here
23     for key in list(self.my_table):
24         if time.time()-self.my_table[key][1]>=10:
25             self.my_table.pop(key)
26     arp=packet.get_header(Arp)
27     if arp is None:
28         log_info("not a Arp packet")
29     else:
30         if arp.operation==1:
31             self.my_table[arp.senderprotoaddr]=[arp.senderhwaddr,time.time()]
32             for inft in self.my_inft:
33                 if arp.targetprotoaddr==inft.ipaddr:
34                     reply_pkt=create_ip_arp_reply(inft.ethaddr,arp.senderhwaddr,inft.ipaddr,arp.senderprotoaddr)
35                     self.net.send_packet(inft,reply_pkt)
36             else:
37                 log_info("operation =2,it is a reply_packet")
38     ...
39
40 def start(self):
41     """A running daemon of the router.
42     Receive packets until the end of time.
43     """
44     while True:
45         try:
46             rcv = self.net.rcv_packet(timeout=1.0)
47         except NoPackets:
48             continue
49         except Shutdown:
50             break
51
52         self.handle_packet(rcv)
53
54     self.stop()
55
56 def stop(self):
57     self.net.shutdown()
58

```

步骤二：

```

14:24:57 2023/10/21 INFO Starting test scenario testcases/myrouter1_testscenario.srpy myrouter.py
14:24:57 2023/10/21 INFO IP addr:192.168.1.100 MAC addr:30:00:00:00:00:01 is added or refreshed
14:24:57 2023/10/21 INFO -----MAC_LIST-----
14:24:57 2023/10/21 INFO IP addr:192.168.1.100 MAC addr:30:00:00:00:00:01 tline:1697869497.4062169
14:24:57 2023/10/21 INFO not a Arp packet
14:24:57 2023/10/21 INFO IP addr:10.10.1.1 MAC addr:60:00:de:ad:be:ef is added or refreshed
14:24:57 2023/10/21 INFO -----MAC_LIST-----
14:24:57 2023/10/21 INFO IP addr:192.168.1.100 MAC addr:30:00:00:00:00:01 tline:1697869497.4062169
14:24:57 2023/10/21 INFO IP addr:10.10.1.1 MAC addr:60:00:de:ad:be:ef tline:1697869497.4068277
14:24:57 2023/10/21 INFO IP addr:10.10.5.5 MAC addr:70:00:ca:fe:c0:de is added or refreshed
14:24:57 2023/10/21 INFO -----MAC_LIST-----
14:24:57 2023/10/21 INFO IP addr:192.168.1.100 MAC addr:30:00:00:00:00:01 tline:1697869497.4062169
14:24:57 2023/10/21 INFO IP addr:10.10.1.1 MAC addr:60:00:de:ad:be:ef tline:1697869497.4068277
14:24:57 2023/10/21 INFO IP addr:10.10.5.5 MAC addr:70:00:ca:fe:c0:de tline:1697869497.407139

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

```

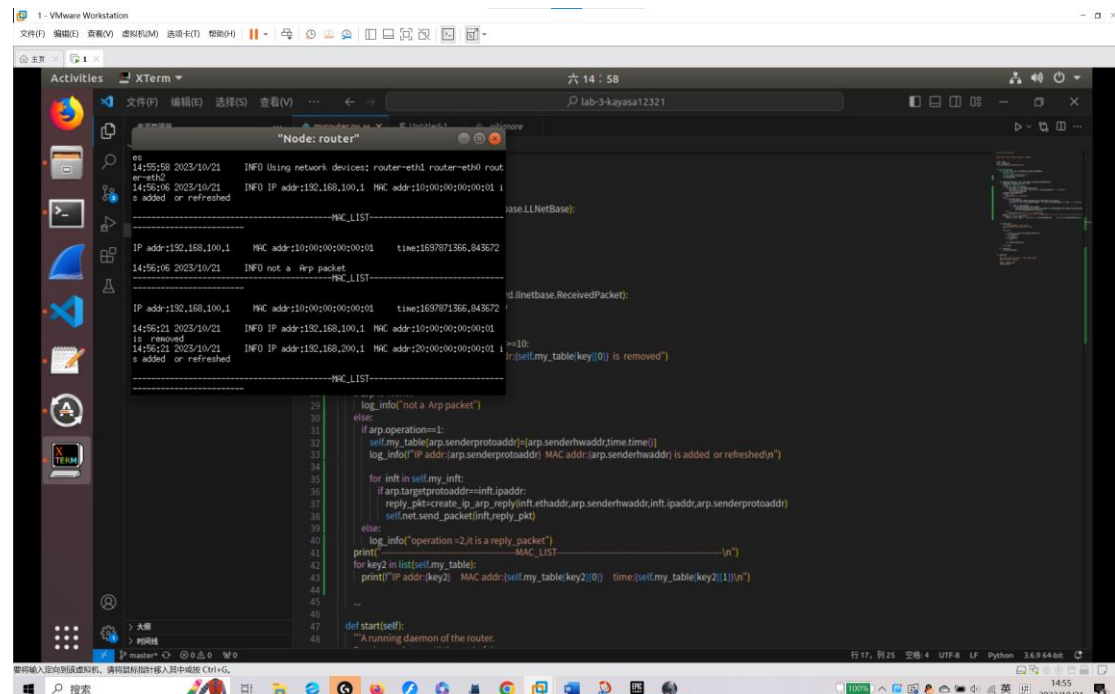
测试样例 1, 2 均为从 192.168.1.100 发送 ARP 请求到接口，在第一次时将其加入，第二次时将其的 time 更新。

测试样例 3 则为不是 arp 请求，所以不做处理也没有 ot。

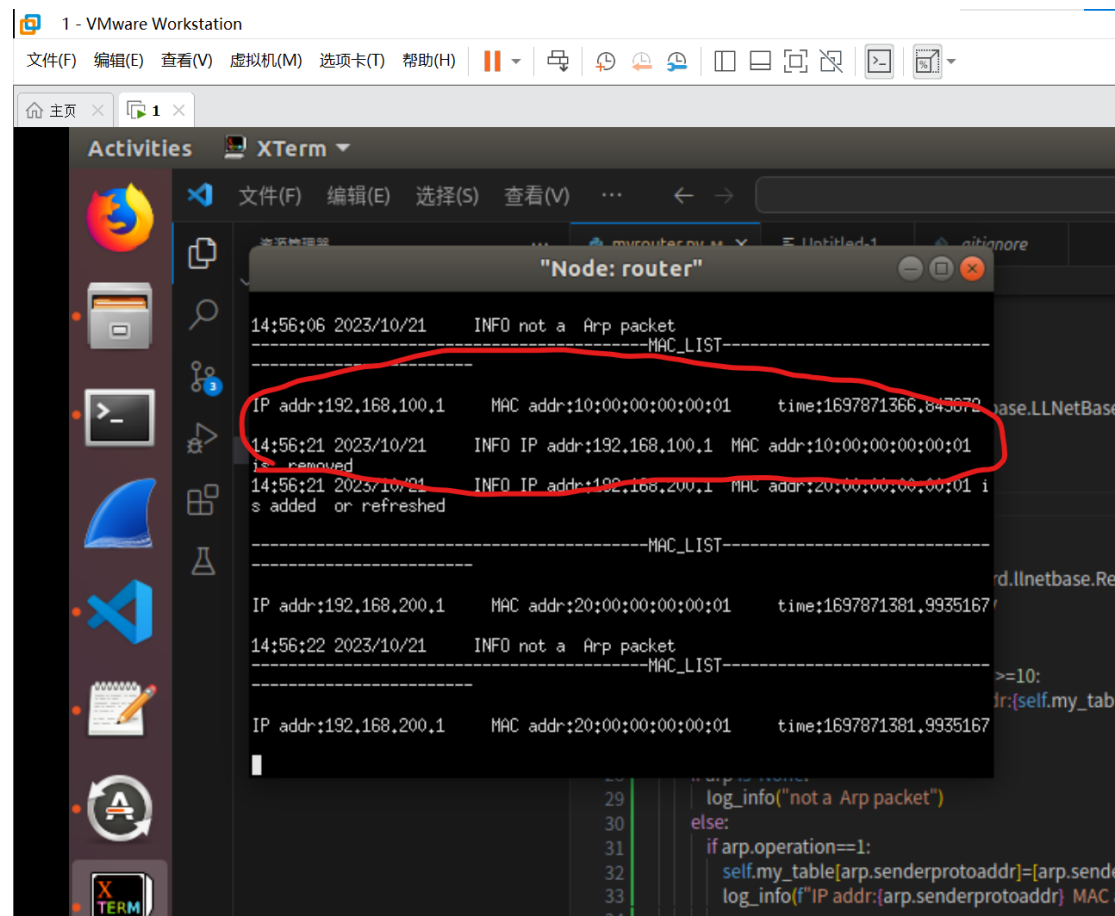
测试样例 4 则是目标地址不在 router 里面也不做处理

测试样例 5, 6;表明从接口一接收到发给自己的 arp 请求并且做出相应的回应，此时将发送方加入缓存表，如图中 mac 表更新。

如果要测试超时机制，我们需要的是采用 ping 操作
先在 server1 ping 到自己的接口，然后从 server2 十秒之后 ping 一下，从 router 的终端来看。



```
14:55:58 2023/10/21 INFO Using network devices: router-eth1 router-eth0 router-eth2
14:56:06 2023/10/21 INFO IP addr:192.168,100,1 MAC addr:10:00:00:00:00:01 is added or refreshed
-----MAC_LIST-----
IP addr:192.168,100,1 MAC addr:10:00:00:00:00:01 time:1697871366,843672
14:56:06 2023/10/21 INFO not a Arp packet
-----MAC_LIST-----
IP addr:192.168,100,1 MAC addr:10:00:00:00:00:01 time:1697871366,843672
14:56:21 2023/10/21 INFO IP addr:192.168,100,1 MAC addr:10:00:00:00:00:01 is removed
14:56:21 2023/10/21 INFO IP addr:192.168,200,1 MAC addr:20:00:00:00:00:01 is added or refreshed
-----MAC_LIST-----
IP addr:192.168,200,1 MAC addr:20:00:00:00:00:01 time:1697871381,9935167
14:56:22 2023/10/21 INFO not a Arp packet
-----MAC_LIST-----
IP addr:192.168,200,1 MAC addr:20:00:00:00:00:01 time:1697871381,9935167
```



```
14:56:06 2023/10/21 INFO not a Arp packet
-----MAC_LIST-----
IP addr:192.168,100,1 MAC addr:10:00:00:00:00:01 time:1697871366,843672
14:56:21 2023/10/21 INFO IP addr:192.168,100,1 MAC addr:10:00:00:00:00:01 is removed
14:56:21 2023/10/21 INFO IP addr:192.168,200,1 MAC addr:20:00:00:00:00:01 is added or refreshed
-----MAC_LIST-----
IP addr:192.168,200,1 MAC addr:20:00:00:00:00:01 time:1697871381,9935167
14:56:22 2023/10/21 INFO not a Arp packet
-----MAC_LIST-----
IP addr:192.168,200,1 MAC addr:20:00:00:00:00:01 time:1697871381,9935167
```

