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：lzh、lsp、wcx

学院	**计算机科学与技术系**	专业（方向）	计算机科学与技术系
学号	211220027	姓名	王秋博
Email	211220027@nju.edu.cn	开始/完成日期	9.30/10.13

1. 实验名称

Lab1: Learning Switch

2. 实验目的

学习网络中的自学习交换机的工作原理并实现不同的工作机制。

3. 实验内容

Task2 Basic Switch

Coding：加入字典 table，eth 和 intf 对应 key 和 value 进行判断即可，在此基础上的后续几个 switch 也都基于此进行排序等操作，以后不再赘述，主要讨论交换机转发逻辑

Deploying：

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	30:00:00:00:00:01	Broadcast	ARP	42	Who has 192.168.100.1? Tell 192.168.100.3
2	0.106066568	Private_00:00:01	30:00:00:00:00:01	ARP	42	192.168.100.1 is at 10:00:00:00:00:01
3	0.651849138	192.168.100.3	192.168.100.1	ICMP	98	Echo (ping) request id=0x3cf4, seq=1/256, ttl=64 (rep
4	0.754046853	192.168.100.1	192.168.100.3	ICMP	98	Echo (ping) reply id=0x3cf4, seq=1/256, ttl=64 (req
5	1.069796486	192.168.100.3	192.168.100.1	ICMP	98	Echo (ping) request id=0x3cf4, seq=2/512, ttl=64 (rep
6	1.170980916	192.168.100.1	192.168.100.3	ICMP	98	Echo (ping) reply id=0x3cf4, seq=2/512, ttl=64 (req
7	5.875978057	Private_00:00:01	30:00:00:00:00:01	ARP	42	Who has 192.168.100.3? Tell 192.168.100.1
8	6.340918925	30:00:00:00:00:01	Private_00:00:01	ARP	42	192.168.100.3 is at 30:00:00:00:00:01

Packet1：初始转发表为空，需先通过 ARP 协议在网络中广播询问 server1 的 MAC 地址。此时，switch 转发表学习 client 的 MAC-端口号的映射；

Packet2：Server1 监听到该 ARP 包并且将自己的 MAC 地址通过 ARP 协议发送给 client，由于之前 switch 转发表已经记录过 client 的 MAC 地址，故可以直接通过已知端口转发。同时 switch 转发表也会学习 server1 的 MAC 地址-端口号的映射。

Packet3-6：client 节点收到 server1 的 ARP 包后，就可以通过已知的 server1 的 MAC 地址生成 ICMP 包通过 switch 发送给 server1 进行 ping 而 server1 收到 client 的 ping 请求，同样通过 switch 发送给 client ping 回复 ICMP 包。Packet5、6 同 3，4。

Packet7，8 则是定时确认 client 的 MAC 地址，以防网络拓扑结构改变。

对 server2 节点抓包,抓包结果如下：

Apply a display filter ... <Ctrl-/>　Expression...

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	30:00:00:00:00:01	Broadcast	ARP	42	Who has 192.168.100.1? Tell 192.168.100.3

而 server2 收到的唯一一个 ICMP 包是第一次 client 询问 server1 的 MAC 地址

广播发送的 ARP 包。

Task3 Time Out

Testing:

```
(syenv) njucs@njucs-VirtualBox:~/workspace/lab-2-Dexter2008$ swyard -t testcases/myswitch_to_testscenario.srpy myswitch_to.py
04:05:14 2023/10/13     INFO Starting test scenario testcases/myswitch_to_testscenario.srpy
04:05:14 2023/10/13     INFO Flooding packet Ethernet 30:00:00:00:00:02->ff:ff:ff:ff:ff:ff IP | IPv4 172.16.42.2->255.255.255.255 ICMP | ICMP EchoRequest 0 0 (0 data bytes) to eth0
04:05:14 2023/10/13     INFO Flooding packet Ethernet 30:00:00:00:00:02->ff:ff:ff:ff:ff:ff IP | IPv4 172.16.42.2->255.255.255.255 ICMP | ICMP EchoRequest 0 0 (0 data bytes) to eth2
04:06:15 2023/10/13     INFO Flooding packet Ethernet 20:00:00:00:00:01->30:00:00:00:00:02 IP | IPv4 192.168.1.100->172.16.42.2 ICMP | ICMP EchoRequest 0 0 (0 data bytes) to eth1
04:06:15 2023/10/13     INFO Flooding packet Ethernet 20:00:00:00:00:01->30:00:00:00:00:02 IP | IPv4 192.168.1.100->172.16.42.2 ICMP | ICMP EchoRequest 0 0 (0 data bytes) to eth2
04:06:15 2023/10/13     INFO Received a packet intended for me

Results for test scenario switch tests: 9 passed, 0 failed, 0 pending


Passed:
1   An Ethernet frame with a broadcast destination address
    should arrive on eth1
2   The Ethernet frame with a broadcast destination address
    should be forwarded out ports eth0 and eth0 and eth2
3   An Ethernet frame from 20:00:00:00:00:01 to
    30:00:00:00:00:02 should arrive on eth0
4   Ethernet frame destined for 30:00:00:00:00:02 should arrive
    on eth1 after self-learning
5   Timeout for 60s
6   An Ethernet frame from 20:00:00:00:00:01 to
    30:00:00:00:00:02 should arrive on eth0
7   Ethernet frame destined for 30:00:00:00:00:02 should be
    flooded out eth1 and eth2
8   An Ethernet frame should arrive on eth2 with destination
    address the same as eth2's MAC address
9   The hub should not do anything in response to a frame
    arriving with a destination address referring to the hub
    itself.
```

Deploying:

可以考虑在 client 节点连续进行两次 ping server1，再间隔 t>10s 后再进行一次 ping 并在 server1 和 server2 节点进行抓包。以形成对比。

Server1 的抓包如下

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	30:00:00:00:00:01	Broadcast	ARP	42	Who has 192.168.100.1? Tell 192.168.100.3
2	0.101697295	Private_00:00:01	30:00:00:00:00:01	ARP	42	192.168.100.1 is at 10:00:00:00:00:01
3	0.441460957	192.168.100.3	192.168.100.1	ICMP	98	Echo (ping) request id=0x397f, seq=1/256, ttl=64 (rep
4	0.543863673	192.168.100.1	192.168.100.3	ICMP	98	Echo (ping) reply id=0x397f, seq=1/256, ttl=64 (rec
5	4.282794506	192.168.100.3	192.168.100.1	ICMP	98	Echo (ping) request id=0x3980, seq=1/256, ttl=64 (rep
6	4.384249603	192.168.100.1	192.168.100.3	ICMP	98	Echo (ping) reply id=0x3980, seq=1/256, ttl=64 (rec
7	5.617006996	Private_00:00:01	30:00:00:00:00:01	ARP	42	Who has 192.168.100.3? Tell 192.168.100.1
8	6.011910447	30:00:00:00:00:01	Private_00:00:01	ARP	42	192.168.100.3 is at 30:00:00:00:00:01
9	71.182770181	192.168.100.3	192.168.100.1	ICMP	98	Echo (ping) request id=0x39b0, seq=1/256, ttl=64 (rep
10	71.289791738	192.168.100.1	192.168.100.3	ICMP	98	Echo (ping) reply id=0x39b0, seq=1/256, ttl=64 (rec
11	76.434490616	30:00:00:00:00:01	Private_00:00:01	ARP	42	Who has 192.168.100.1? Tell 192.168.100.3
12	76.526796585	Private_00:00:01	30:00:00:00:00:01	ARP	42	Who has 192.168.100.3? Tell 192.168.100.1
13	76.535562741	Private_00:00:01	30:00:00:00:00:01	ARP	42	192.168.100.1 is at 10:00:00:00:00:01
14	76.854498317	30:00:00:00:00:01	Private_00:00:01	ARP	42	192.168.100.3 is at 30:00:00:00:00:01

前四条数据与 task2 的前四条相同，由于第二次 ping 时间间隔小于 10s，client 和 server1 的 mac 地址映射仍存在于转发表中，故 5，6 条数据包仍以 ICMP 包形式点对点传输。7.8 条为定时确认。而第三次 ping 时，时间间隔大于 10，转发表内映射已被删除，则需要重新广播。

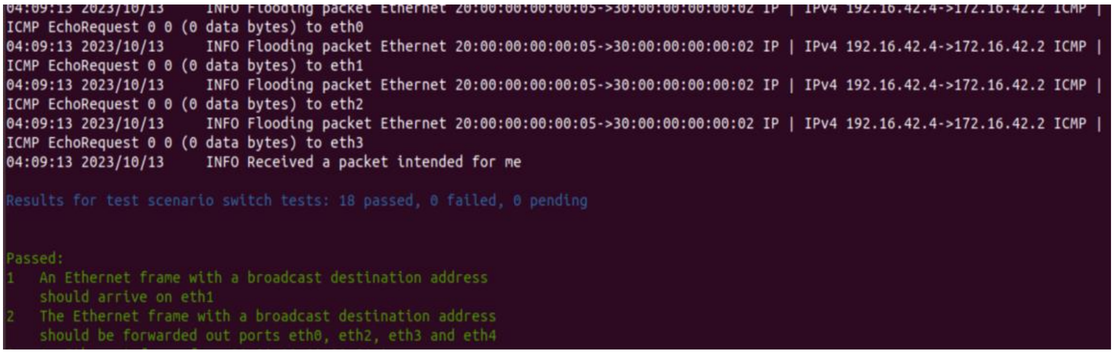
Apply a display filter ... <Ctrl-/> Expression...

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	30:00:00:00:00:01	Broadcast	ARP	42	Who has 192.168.100.1? Tell 192.168.100.3
2	71.182769317	192.168.100.3	192.168.100.1	ICMP	98	Echo (ping) request id=0x39b0, seq=1/256, ttl=64 (no

而对于 server2，第一次 client 广播时，发送一个 ARP 包来询问 server2 而第二次 ping 时为点对点传输，不会收到数据包，第三次 ping 时再次收到 client 的再次广播 ICMP 包。

Task4 Least Recently Used

Testing:



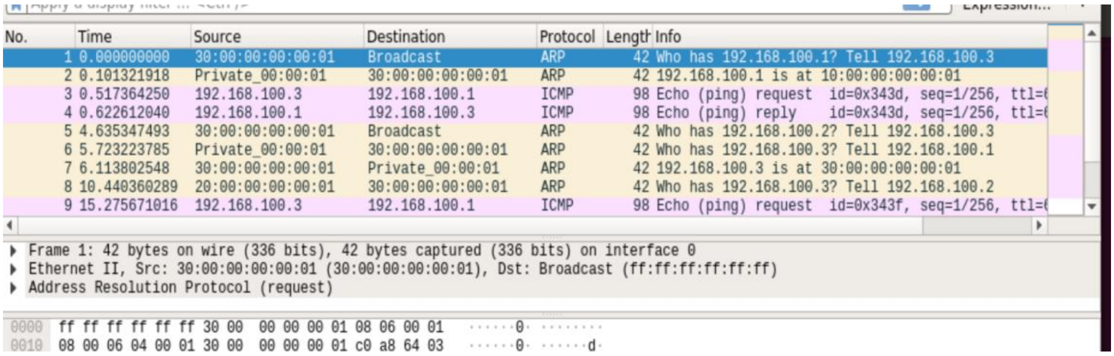
```
04:09:13 2023/10/13     INFO Flooding packet Ethernet 20:00:00:00:00:05->30:00:00:00:00:02 IP | IPv4 192.16.42.4->172.16.42.2 ICMP |
ICMP EchoRequest 0 0 (0 data bytes) to eth0
04:09:13 2023/10/13     INFO Flooding packet Ethernet 20:00:00:00:00:05->30:00:00:00:00:02 IP | IPv4 192.16.42.4->172.16.42.2 ICMP |
ICMP EchoRequest 0 0 (0 data bytes) to eth1
04:09:13 2023/10/13     INFO Flooding packet Ethernet 20:00:00:00:00:05->30:00:00:00:00:02 IP | IPv4 192.16.42.4->172.16.42.2 ICMP |
ICMP EchoRequest 0 0 (0 data bytes) to eth2
04:09:13 2023/10/13     INFO Flooding packet Ethernet 20:00:00:00:00:05->30:00:00:00:00:02 IP | IPv4 192.16.42.4->172.16.42.2 ICMP |
ICMP EchoRequest 0 0 (0 data bytes) to eth3
04:09:13 2023/10/13     INFO Received a packet intended for me

Results for test scenario switch tests: 18 passed, 0 failed, 0 pending


Passed:
1   An Ethernet frame with a broadcast destination address
    should arrive on eth1
2   The Ethernet frame with a broadcast destination address
    should be forwarded out ports eth0, eth2, eth3 and eth4
```

Deploying：考虑到若转发表大小设成 5，对于该 3 节点的拓扑结构则 lru 没有意义，故临时将转发表大小设为 2。

测试采用先各 ping 一次 server1 和 server2，后 ping 两次 server1；

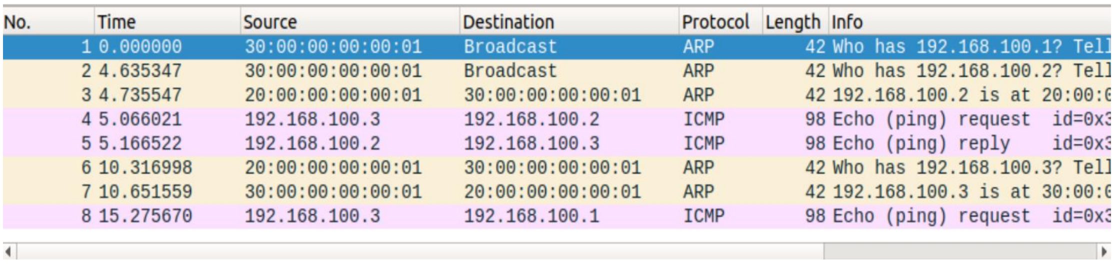


No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	30:00:00:00:00:01	Broadcast	ARP	42	Who has 192.168.100.1? Tell 192.168.100.3
2	0.101321918	Private_00:00:01	30:00:00:00:00:01	ARP	42	192.168.100.1 is at 10:00:00:00:00:01
3	0.517364250	192.168.100.3	192.168.100.1	ICMP	98	Echo (ping) request id=0x343d, seq=1/256, ttl=6
4	0.622612040	192.168.100.1	192.168.100.3	ICMP	98	Echo (ping) reply id=0x343d, seq=1/256, ttl=6
5	4.635347493	30:00:00:00:00:01	Broadcast	ARP	42	Who has 192.168.100.2? Tell 192.168.100.3
6	5.723223785	Private_00:00:01	30:00:00:00:00:01	ARP	42	Who has 192.168.100.3? Tell 192.168.100.1
7	6.113802548	30:00:00:00:00:01	Private_00:00:01	ARP	42	192.168.100.3 is at 30:00:00:00:00:01
8	10.440360289	20:00:00:00:00:01	30:00:00:00:00:01	ARP	42	Who has 192.168.100.3? Tell 192.168.100.2
9	15.275671016	192.168.100.3	192.168.100.1	ICMP	98	Echo (ping) request id=0x343f, seq=1/256, ttl=6

```
Frame 1: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0
Ethernet II, Src: 30:00:00:00:00:01 (30:00:00:00:00:01), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
Address Resolution Protocol (request)

0000  ff ff ff ff ff ff 30 00  00 00 00 01 08 06 00 01   ......0. ........
0010  08 00 06 04 00 01 30 00  00 00 00 01 c0 a8 64 03   ......0. ......d.
```

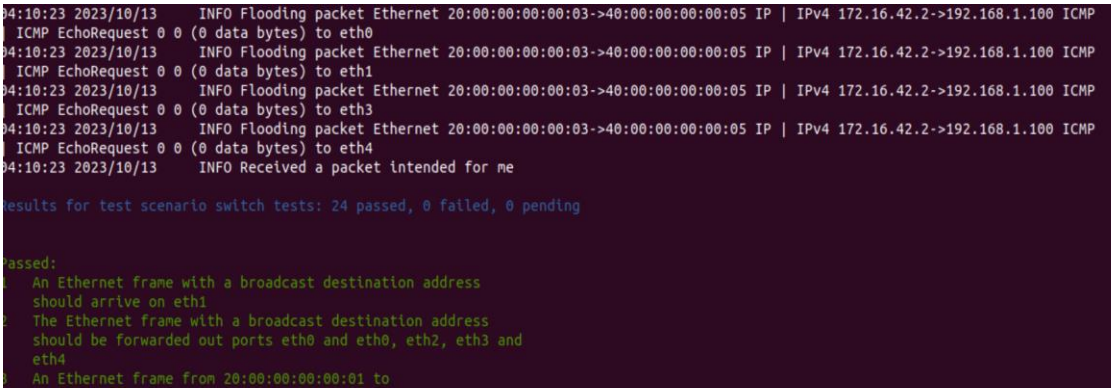
Server1 前 4 个包不再阐述，此时转发表记住了 client 和 server1 的 mac 地址，而 ping server2 后 client 开始广播寻找 server2,而 client 地址在表中，故 server2 点对点回复 ARP 包（server2.3）此时转发表内 server1 地址被删除更换为 server2，然后进行点对点的 ping server2 操作。后 server1 发送 ARP 给 client 确认行为，server2 再次被替换为 server1，之后进行 ping server1 行为，转发表不发生变化。随后一次 ping server1 时，server2 再次发送 ARP 包确认 client，导致转发表再次发生变化，ping server1 时 client flood out 导致了 server2 收到 ICMP 包（server2.8）



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	30:00:00:00:00:01	Broadcast	ARP	42	Who has 192.168.100.1? Tell
2	4.635347	30:00:00:00:00:01	Broadcast	ARP	42	Who has 192.168.100.2? Tell
3	4.735547	20:00:00:00:00:01	30:00:00:00:00:01	ARP	42	192.168.100.2 is at 20:00:0
4	5.066021	192.168.100.3	192.168.100.2	ICMP	98	Echo (ping) request id=0x3
5	5.166522	192.168.100.2	192.168.100.3	ICMP	98	Echo (ping) reply id=0x3
6	10.316998	20:00:00:00:00:01	30:00:00:00:00:01	ARP	42	Who has 192.168.100.3? Tell
7	10.651559	30:00:00:00:00:01	20:00:00:00:00:01	ARP	42	192.168.100.3 is at 30:00:0
8	15.275670	192.168.100.3	192.168.100.1	ICMP	98	Echo (ping) request id=0x3

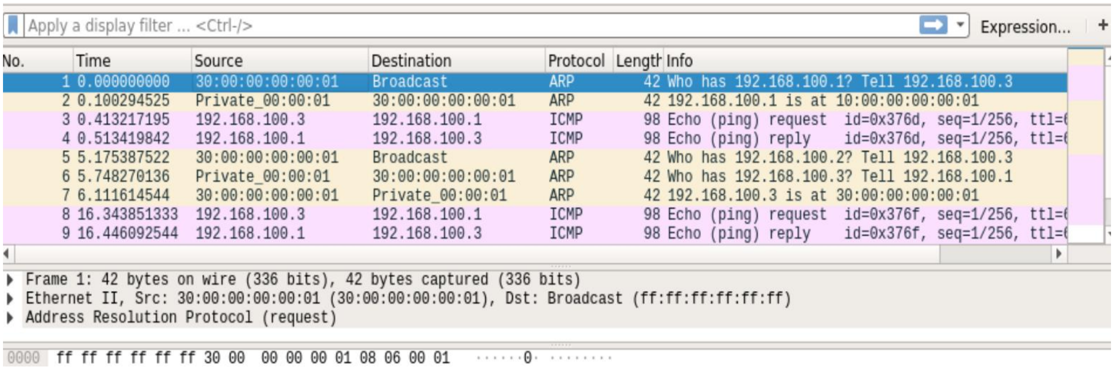
Task5 Least Traffic Volume

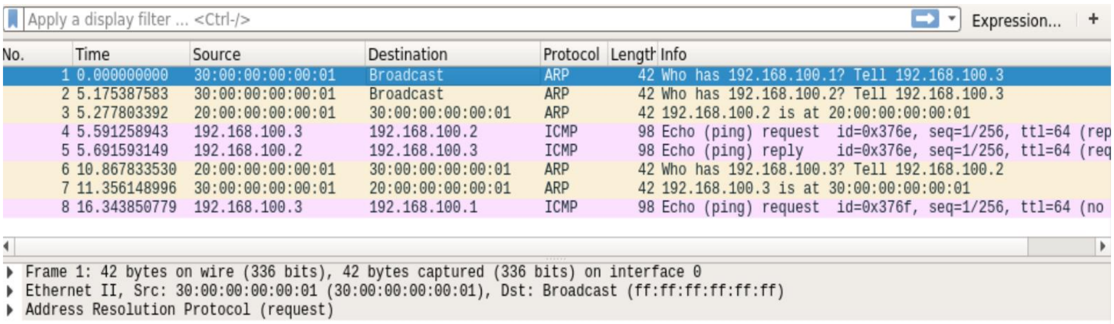
Testing:



Deploying：与 lru 一样，临时设置转发表大小为 2

实践操作也相同，server1 和 server2 各 ping 一次，后 ping 两次 server1





前四个包仍不再赘述，为第一次 ping server1，因为广播不算流量此时转发表内为{client：2|server1：1}后 ping server2，client 进行广播询问，server2 发 ARP 包进行回复，此时 server1 被替换{client：4|server2：1}，在第二次 ping server1 前，server1 先发 ARP 包确认 client，client 回复 server1，故 server2 被替换{client：5|server1：1}接下来 ping server1，点对点传输，转发表变为{client：6|server1：2}；再最后一次 ping server1 前，server2 确认 client 导致 server1 再次被替换{client：7|server2：1}最后 ping server1{client：8|server1：1}

4. 实验结果

本节实验结果基本于实验过程中阐述，不再赘述

5. 核心代码

同实验结果

6. 总结与感想

通过本次实验，深入了解了 switch 的逻辑，强化了 wireshark 的使用。实验确实巧妙且形象，感谢手册编写人员，也仍希望以后实验顺利。