

گزارش کار جلسه سوم

محمد صالح پژند - زینب باقیان - امیر محمد درپوش

(کد تغییر یافته):

```
1 #!/usr/bin/python
2
3 from mininet.net import Mininet
4 from mininet.node import Controller
5 from mininet.cli import CLI
6 from mininet.log import setLogLevel, info
7
8
9 def firstNetwork():
10     net = Mininet()
11     info('*** Adding controller\n')
12     net.addController('c0')
13
14     info('*** Adding hosts\n')
15     h1 = net.addHost('h1', ip='10.10.14.1/24')
16     h2 = net.addHost('h2', ip='10.10.24.2/24')
17     h3 = net.addHost('h3', ip='10.10.34.3/24')
18     h4 = net.addHost('h4', ip='10.10.14.4/24')
19
20     info('*** Adding switch\n')
21     s14 = net.addSwitch('s14')
22     s24 = net.addSwitch('s24')
23     s34 = net.addSwitch('s34')
24
25     info('*** Creating links\n')
26     net.addLink(h1, s14, intfName1="h1-eth0", intfName2="s14-eth1")
27     net.addLink(h2, s24, intfName1="h2-eth0", intfName2="s24-eth1")
28     net.addLink(h3, s34, intfName1="h3-eth0", intfName2="s34-eth1")
29     net.addLink(h4, s14, intfName1="h4-eth0", intfName2="s14-eth2")
30     net.addLink(h4, s24, intfName1="h4-eth1", intfName2="s24-eth2")
31     net.addLink(h4, s34, intfName1="h4-eth2", intfName2="s34-eth2")
32
33     h4.cmd('ip addr add 10.10.24.4/24 dev h4-eth1')
```

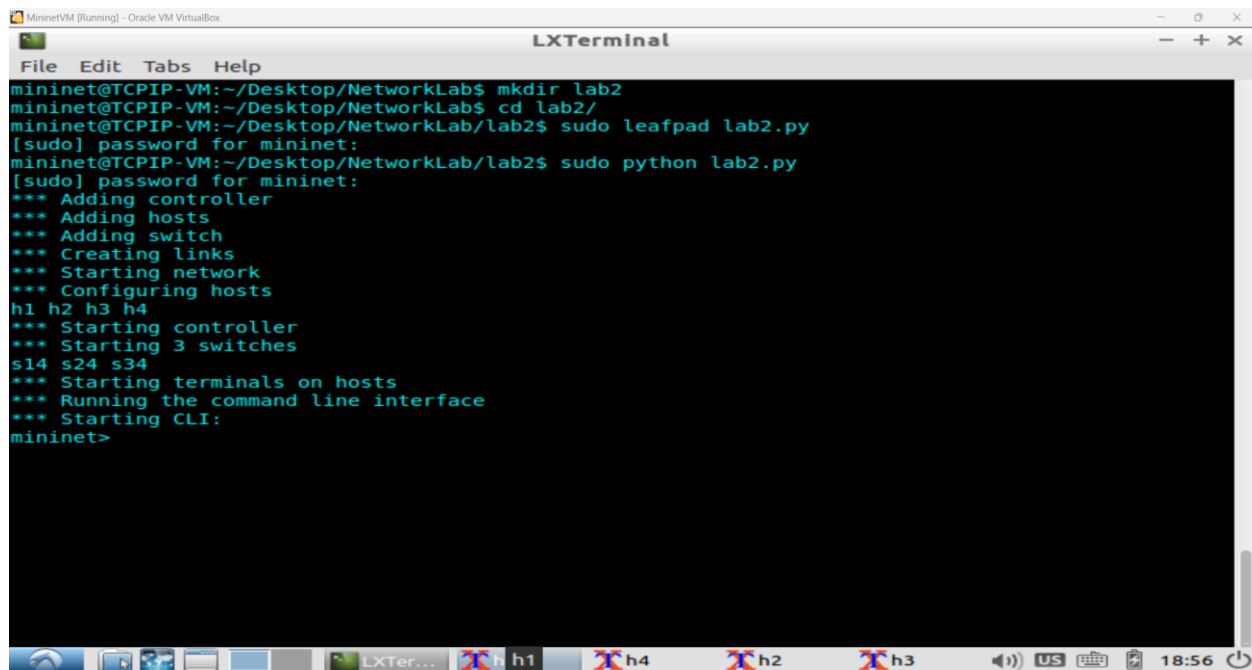
```

34     h4.cmd('ip addr add 10.10.34.4/24 dev h4-eth2')
35     h4.cmd('echo 1 > /proc/sys/net/ipv4/ip_forward')
36
37     info('*** Starting network\n')
38     net.start()
39
40     info('*** Adding Gateways\n')
41     h1.cmd('ip route add default via 10.10.14.4')
42     h2.cmd('ip route add default via 10.10.24.4')
43     h3.cmd('ip route add default via 10.10.34.4')
44
45     # This is used to run commands on the hosts
46     info('*** Starting terminals on hosts\n')
47     h1.cmd('xterm -xrm "XTerm.vt100.allowTitleOps: false" -T h1 &')
48     h2.cmd('xterm -xrm "XTerm.vt100.allowTitleOps: false" -T h2 &')
49     h3.cmd('xterm -xrm "XTerm.vt100.allowTitleOps: false" -T h3 &')
50     h4.cmd('xterm -xrm "XTerm.vt100.allowTitleOps: false" -T h4 &')
51
52     info('*** Running the command line interface\n')
53     CLI(net)
54     info('*** Closing the terminals on the hosts\n')
55     h1.cmd("killall xterm")
56     h2.cmd("killall xterm")
57     h3.cmd("killall xterm")
58     h4.cmd("killall xterm")
59
60     info('*** Stopping network')
61     net.stop()
62
63
64 if __name__ == '__main__':
65     setLogLevel('info')
66     firstNetwork()

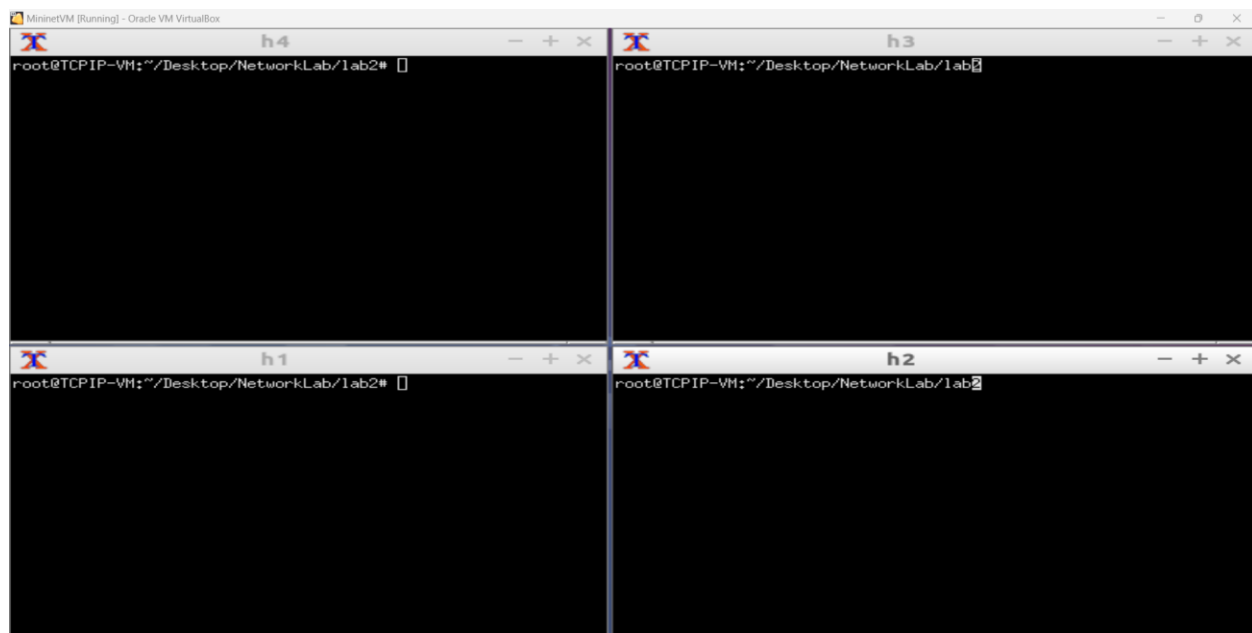
```

سوال 1 و 2-

ابتدا شبکه را تشکیل می‌دهیم



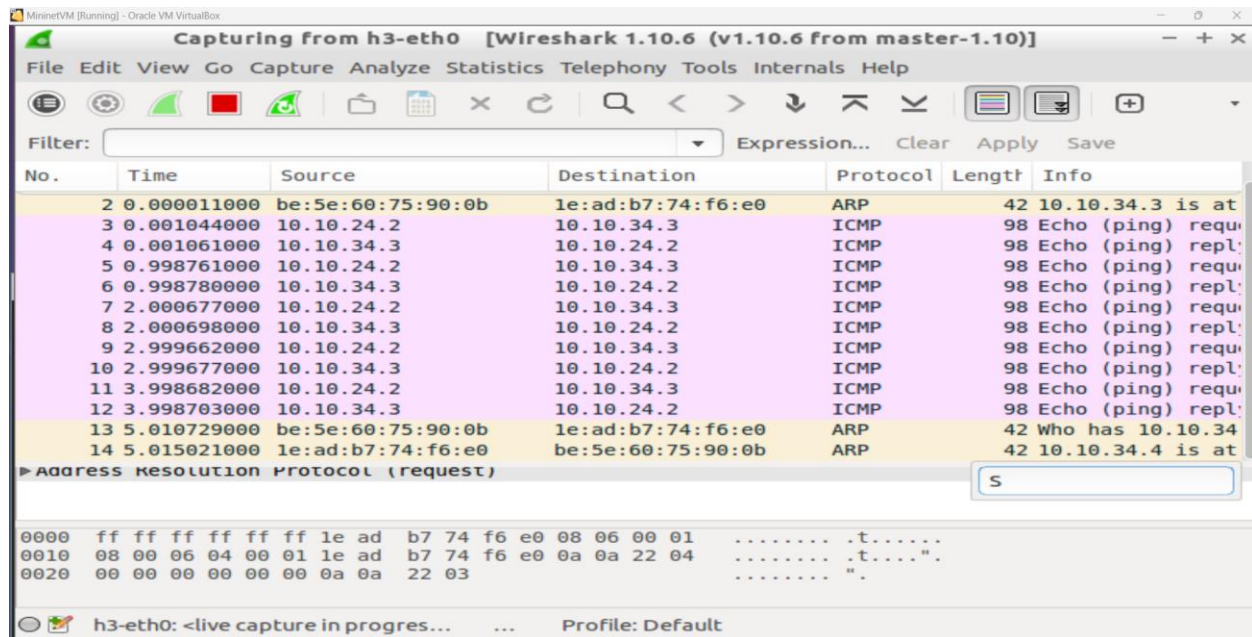
```
MininetVM [Running] - Oracle VM VirtualBox
LXTerminal
File Edit Tabs Help
mininet@TCPIP-VM:~/Desktop/NetworkLab$ mkdir lab2
mininet@TCPIP-VM:~/Desktop/NetworkLab$ cd lab2/
mininet@TCPIP-VM:~/Desktop/NetworkLab/lab2$ sudo leafpad lab2.py
[sudo] password for mininet:
mininet@TCPIP-VM:~/Desktop/NetworkLab/lab2$ sudo python lab2.py
[sudo] password for mininet:
*** Adding controller
*** Adding hosts
*** Adding switch
*** Creating links
*** Starting network
*** Configuring hosts
h1 h2 h3 h4
*** Starting controller
*** Starting 3 switches
s14 s24 s34
*** Starting terminals on hosts
*** Running the command line interface
*** Starting CLI:
mininet>
```



در مرحله اول باید در **h4** کاری کنیم که بسته ها به جای فرستاده شدن به **h1** به **h3** ارسال شوند.

```
h4
root@TCPIP-VM:~/Desktop/NetworkLab/lab2# iptables -t nat -A PREROUTING -p icmp -s 10.10.24.2 -d 10.10.14.1 -j DNAT --to 10.10.34.3
```

```
h2
root@TCPIP-VM:~/Desktop/NetworkLab/lab2# ping 10.10.14.1
PING 10.10.14.1 (10.10.14.1) 56(84) bytes of data:
64 bytes from 10.10.14.1: icmp_seq=1 ttl=63 time=3.65 ms
64 bytes from 10.10.14.1: icmp_seq=2 ttl=63 time=2.12 ms
64 bytes from 10.10.14.1: icmp_seq=3 ttl=63 time=0.095 ms
64 bytes from 10.10.14.1: icmp_seq=4 ttl=63 time=0.142 ms
64 bytes from 10.10.14.1: icmp_seq=5 ttl=63 time=0.092 ms
^C
--- 10.10.14.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4021ms
rtt min/avg/max/mdev = 0.092/1.221/3.655/1.445 ms
root@TCPIP-VM:~/Desktop/NetworkLab/lab2#
```



تا این مرحله توانستیم مسیر بسته هارا به سمت خودمان عوض کنیم. حالا باید بسته هارا دوباره به **h1** هدایت کنیم به طوری که مشخص نشود که این بسته قبلا در **h3** بوده. برای اینکار مقصد را به **h1** تغییر میدهیم ولی در بین راه دوباره **h4** ان هارا به **h3** برمیگرداند . برای حل این مشکل مقصد این بسته هارا نیز به **h4** تغییر میدهیم. برای اینکار دستورات زیر را اجرا میکنیم:

```
root@TCPIP-VM:~/Desktop/NetworkLab/lab2# iptables -t nat -A POSTROUTING -p icmp -s 10.10.24.2 -d 10.10.14.1 -j SNAT --to 10.10.34.3
root@TCPIP-VM:~/Desktop/NetworkLab/lab2# iptables -t nat -A PREROUTING -p icmp -s 10.10.24.2 -d 10.10.34.3 -j DNAT --to 10.10.14.1
root@TCPIP-VM:~/Desktop/NetworkLab/lab2#
```

یکبار دیگر از **h2** پینگ میکنیم و در نتیجه داریم:

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	de:c4:7d:30:4c:bb	Broadcast	ARP	42	Who has 10.10.14.1? Tell 10.10.14.4
2	0.000007407	0e:04:19:da:07:7d	de:c4:7d:30:4c:bb	ARP	42	10.10.14.1 is at 0e:04:19:da:07:7d
3	0.000391161	10.10.34.3	10.10.14.1	ICMP	98	Echo (ping) request id=0x0615, seq=1/256, ttl=61 (r
4	0.000401487	10.10.14.1	10.10.34.3	ICMP	98	Echo (ping) reply id=0x0615, seq=1/256, ttl=64 (r
5	1.003430080	10.10.34.3	10.10.14.1	ICMP	98	Echo (ping) request id=0x0615, seq=2/512, ttl=61 (r
6	1.003461801	10.10.14.1	10.10.34.3	ICMP	98	Echo (ping) reply id=0x0615, seq=2/512, ttl=64 (r
7	2.003201607	10.10.34.3	10.10.14.1	ICMP	98	Echo (ping) request id=0x0615, seq=3/768, ttl=61 (r
8	2.003230369	10.10.14.1	10.10.34.3	ICMP	98	Echo (ping) reply id=0x0615, seq=3/768, ttl=64 (r
9	3.011577062	10.10.34.3	10.10.14.1	ICMP	98	Echo (ping) request id=0x0615, seq=4/1024, ttl=61 (
10	3.011602724	10.10.14.1	10.10.34.3	ICMP	98	Echo (ping) reply id=0x0615, seq=4/1024, ttl=64 (
11	4.034786113	10.10.34.3	10.10.14.1	ICMP	98	Echo (ping) request id=0x0615, seq=5/1280, ttl=61 (
12	4.034811974	10.10.14.1	10.10.34.3	ICMP	98	Echo (ping) reply id=0x0615, seq=5/1280, ttl=64 (
13	5.058679252	10.10.34.3	10.10.14.1	ICMP	98	Echo (ping) request id=0x0615, seq=6/1536, ttl=61 (
14	5.058689593	10.10.14.1	10.10.34.3	ICMP	98	Echo (ping) reply id=0x0615, seq=6/1536, ttl=64 (
15	5.090699990	0e:04:19:da:07:7d	de:c4:7d:30:4c:bb	ARP	42	Who has 10.10.14.4? Tell 10.10.14.1
16	5.095164063	de:c4:7d:30:4c:bb	0e:04:19:da:07:7d	ARP	42	10.10.14.4 is at de:c4:7d:30:4c:bb
17	6.086398054	10.10.34.3	10.10.14.1	ICMP	98	Echo (ping) request id=0x0615, seq=7/1792, ttl=61 (
18	6.086423615	10.10.14.1	10.10.34.3	ICMP	98	Echo (ping) reply id=0x0615, seq=7/1792, ttl=64 (
19	7.107107429	10.10.34.3	10.10.14.1	ICMP	98	Echo (ping) request id=0x0615, seq=8/2048, ttl=61 (
20	7.107133962	10.10.14.1	10.10.34.3	ICMP	98	Echo (ping) reply id=0x0615, seq=8/2048, ttl=64 (
21	8.130765425	10.10.34.3	10.10.14.1	ICMP	98	Echo (ping) request id=0x0615, seq=9/2304, ttl=61 (
22	8.130791220	10.10.14.1	10.10.34.3	ICMP	98	Echo (ping) reply id=0x0615, seq=9/2304, ttl=64 (
23	9.154530003	10.10.34.3	10.10.14.1	ICMP	98	Echo (ping) request id=0x0615, seq=10/2560, ttl=61
24	9.154556157	10.10.14.1	10.10.34.3	ICMP	98	Echo (ping) reply id=0x0615, seq=10/2560, ttl=64
25	10.178676036	10.10.34.3	10.10.14.1	ICMP	98	Echo (ping) request id=0x0615, seq=11/2816, ttl=61

تنها مورد باقی مانده این است که در **h1** که مقصد نهایی است چون مبدا دیگر **h2** نیست ممکن است دراپ شود.

برای حل این مشکل مبدا را در **h4** عوض میکنیم. نتیجه نهایی به شکل زیر است:

```

h4
root@TCPIP-VM:~/Desktop/NetworkLab/lab2# iptables -t nat -A PREROUTING -p icmp -s 10.10.24.2 -d 10.10.14.1 -j DNAT --to 10.10.34.3
root@TCPIP-VM:~/Desktop/NetworkLab/lab2# iptables -t nat -A POSTROUTING -p icmp -s 10.10.34.3 -d 10.10.14.1 -j SNAT --to 10.10.24.2
root@TCPIP-VM:~/Desktop/NetworkLab/lab2#

```

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.10.24.2	10.10.14.1	ICMP	98	Echo (ping) request id=0x06ab, seq=1/256, ttl=61 (re
2	0.000009339	10.10.14.1	10.10.24.2	ICMP	98	Echo (ping) reply id=0x06ab, seq=1/256, ttl=64 (re
3	1.002031748	10.10.24.2	10.10.14.1	ICMP	98	Echo (ping) request id=0x06ab, seq=2/512, ttl=61 (re
4	1.002048927	10.10.14.1	10.10.24.2	ICMP	98	Echo (ping) reply id=0x06ab, seq=2/512, ttl=64 (re
5	2.003348313	10.10.24.2	10.10.14.1	ICMP	98	Echo (ping) request id=0x06ab, seq=3/768, ttl=61 (re
6	2.003380695	10.10.14.1	10.10.24.2	ICMP	98	Echo (ping) reply id=0x06ab, seq=3/768, ttl=64 (re
7	3.015654863	10.10.24.2	10.10.14.1	ICMP	98	Echo (ping) request id=0x06ab, seq=4/1024, ttl=61 (r
8	3.015681764	10.10.14.1	10.10.24.2	ICMP	98	Echo (ping) reply id=0x06ab, seq=4/1024, ttl=64 (r
9	4.039957422	10.10.24.2	10.10.14.1	ICMP	98	Echo (ping) request id=0x06ab, seq=5/1280, ttl=61 (r
10	4.039984669	10.10.14.1	10.10.24.2	ICMP	98	Echo (ping) reply id=0x06ab, seq=5/1280, ttl=64 (r
11	5.063702358	10.10.24.2	10.10.14.1	ICMP	98	Echo (ping) request id=0x06ab, seq=6/1536, ttl=61 (r
12	5.063728426	10.10.14.1	10.10.24.2	ICMP	98	Echo (ping) reply id=0x06ab, seq=6/1536, ttl=64 (r
13	5.095620017	0e:04:19:da:07:7d	de:c4:7d:30:4c:bb	ARP	42	Who has 10.10.14.4? Tell 10.10.14.1
14	5.100237424	de:c4:7d:30:4c:bb	0e:04:19:da:07:7d	ARP	42	Who has 10.10.14.1? Tell 10.10.14.4
15	5.100249327	0e:04:19:da:07:7d	de:c4:7d:30:4c:bb	ARP	42	10.10.14.1 is at 0e:04:19:da:07:7d
16	5.102997103	de:c4:7d:30:4c:bb	0e:04:19:da:07:7d	ARP	42	10.10.14.4 is at de:c4:7d:30:4c:bb
17	6.087712537	10.10.24.2	10.10.14.1	ICMP	98	Echo (ping) request id=0x06ab, seq=7/1792, ttl=61 (r
18	6.087739053	10.10.14.1	10.10.24.2	ICMP	98	Echo (ping) reply id=0x06ab, seq=7/1792, ttl=64 (r
19	7.111509975	10.10.24.2	10.10.14.1	ICMP	98	Echo (ping) request id=0x06ab, seq=8/2048, ttl=61 (r
20	7.111534398	10.10.14.1	10.10.24.2	ICMP	98	Echo (ping) reply id=0x06ab, seq=8/2048, ttl=64 (r
21	8.135702773	10.10.24.2	10.10.14.1	ICMP	98	Echo (ping) request id=0x06ab, seq=9/2304, ttl=61 (r
22	8.135729439	10.10.14.1	10.10.24.2	ICMP	98	Echo (ping) reply id=0x06ab, seq=9/2304, ttl=64 (r

سوال 3-

خیر. برای اینکه بسته هارا به خودمان برسانیم باید به شکلی **h4** را تغییر داد ولی اگر **h3** را تغییر ندهیم یا بسته ها در **h3** میمانند یا به شکلی نامعتبر به مقصد میرسند که باعث دراپ شدنشان میشود.

سوال 4-

بله. با توجه به پارامتر **ttl** میتوان فهمید که بسته های دریافتی بیش از حد معمول تاخیر دارند و مشخص میشود که در بین راه مسیر دیگری را طی کرده اند. برای مثال اینجا **ttl** باید **63** باشد و اگر کمتر از این شود یعنی از **h3** هم عبور کرده

