



دانشگاه صنعتی اصفهان
دانشکده مهندسی برق و کامپیوتر

عنوان: پروژه‌ی دوم درس شبکه‌های کامپیوتری ۲

نام و نام خانوادگی: علیرضا ابره فروش

شماره دانشجویی: ۹۸۱۶۶۰۳

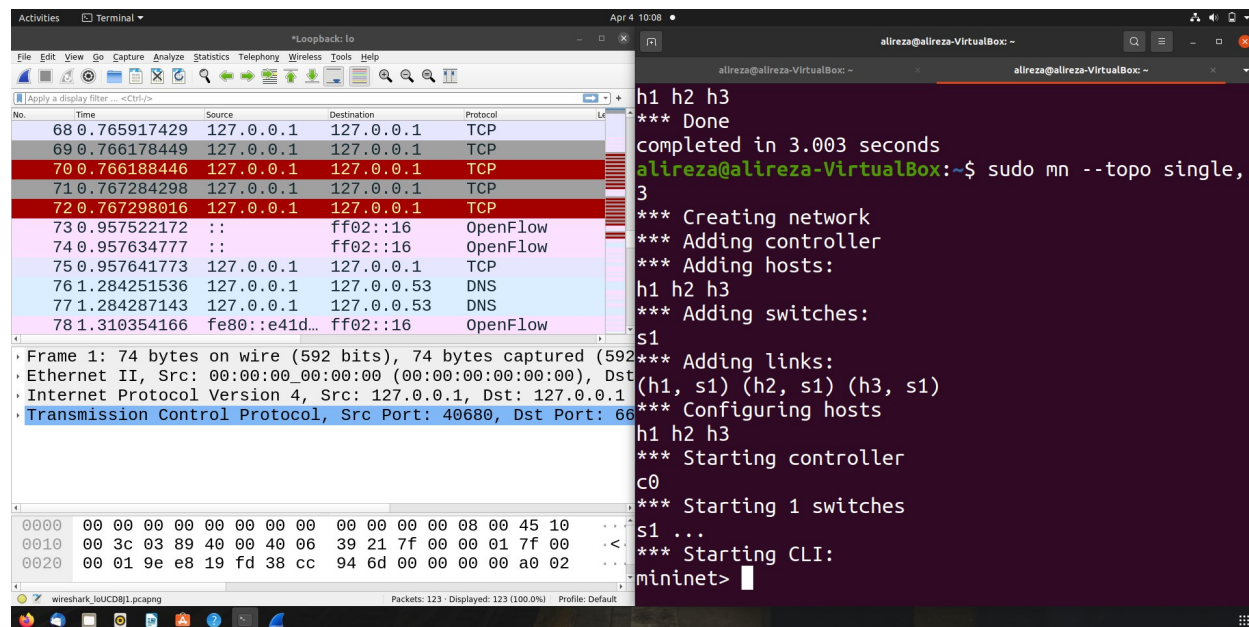
نیم سال تحصیلی: بهار ۱۴۰۱/۱۴۰۰

مدرس: دکتر مسعودرضا هاشمی

۱

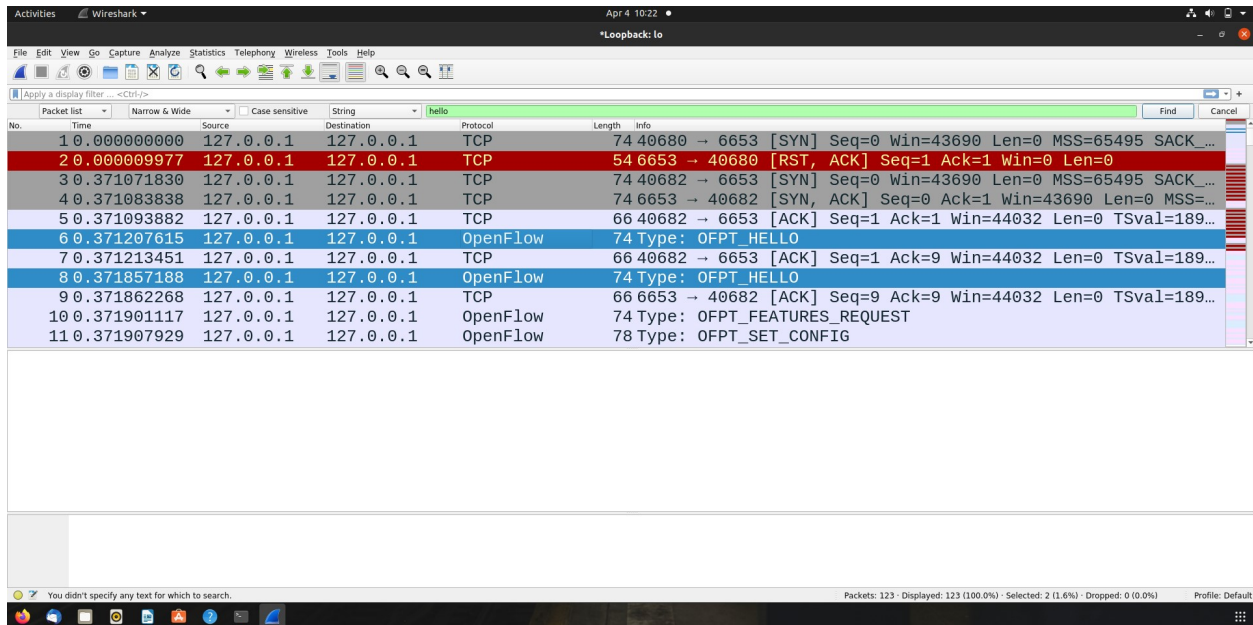
۱.۱

همانطور که در تصویر زیر می‌بینیم، بسته‌هایی که در وایرشارک کپچر شده‌اند از پروتکل‌های TCP، OpenFlow و DNS استفاده کرده‌اند.



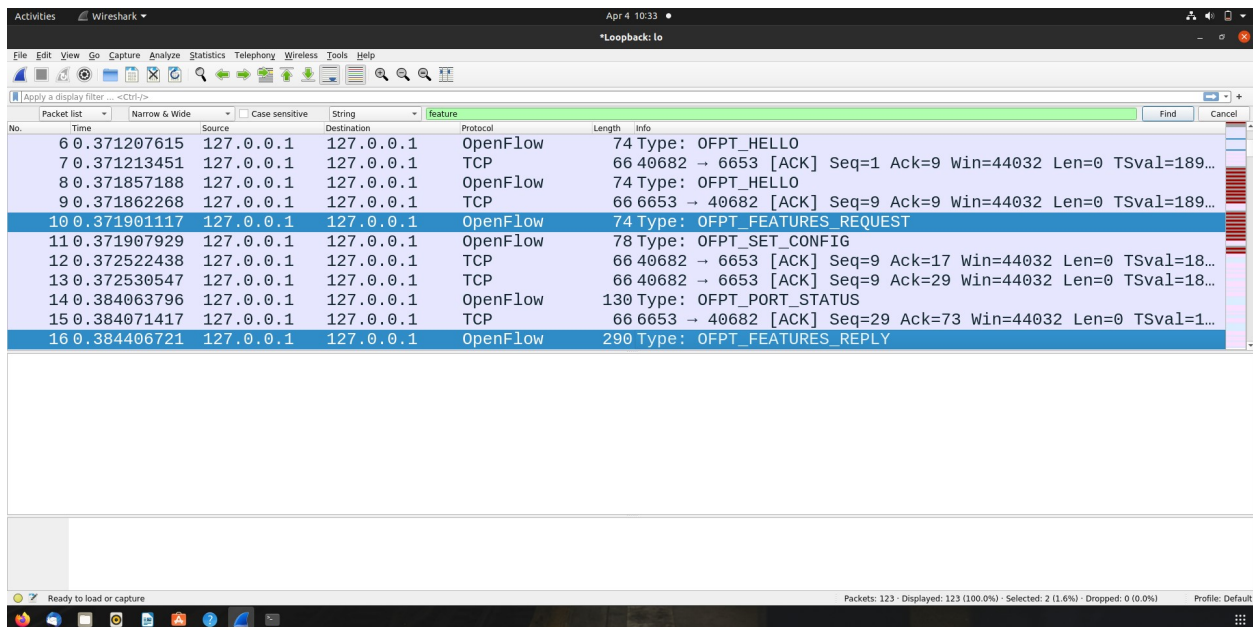
شکل ۱: رصد پکت‌ها در وایرشارک

۲.۱



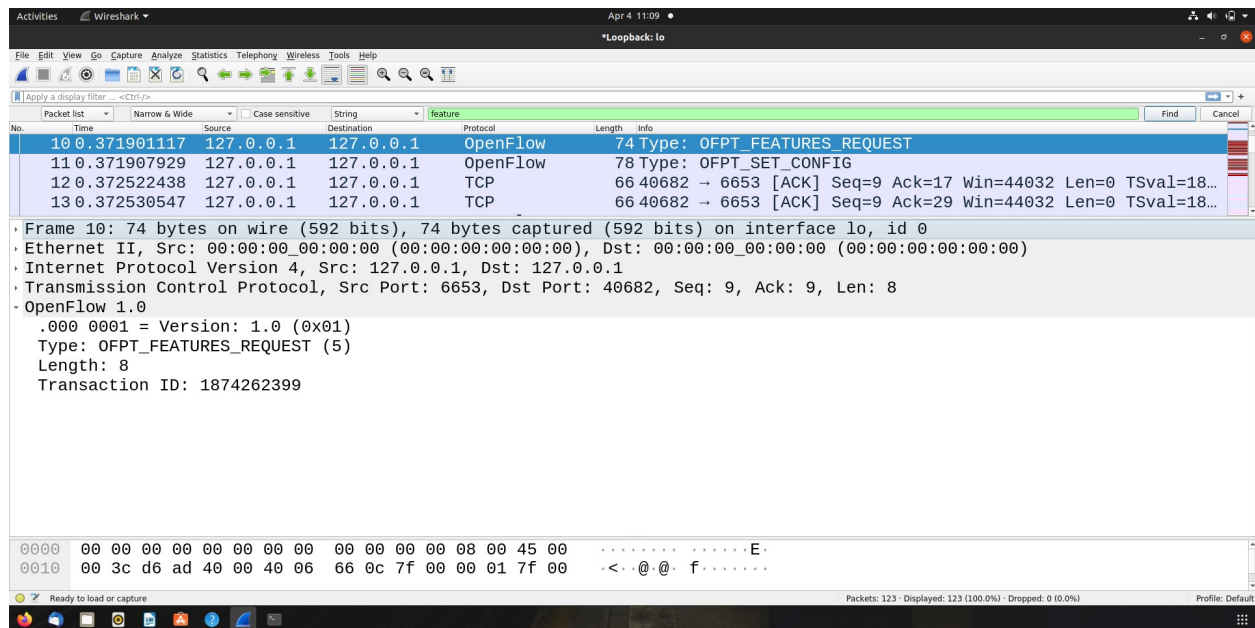
شکل ۲: رد و بدل شدن پیام *Hello* بین سوئیچ و کنترلر در زمان‌های *0.371207615* و *0.371857188* ثانیه پس از اجرای دستور

۳.۱



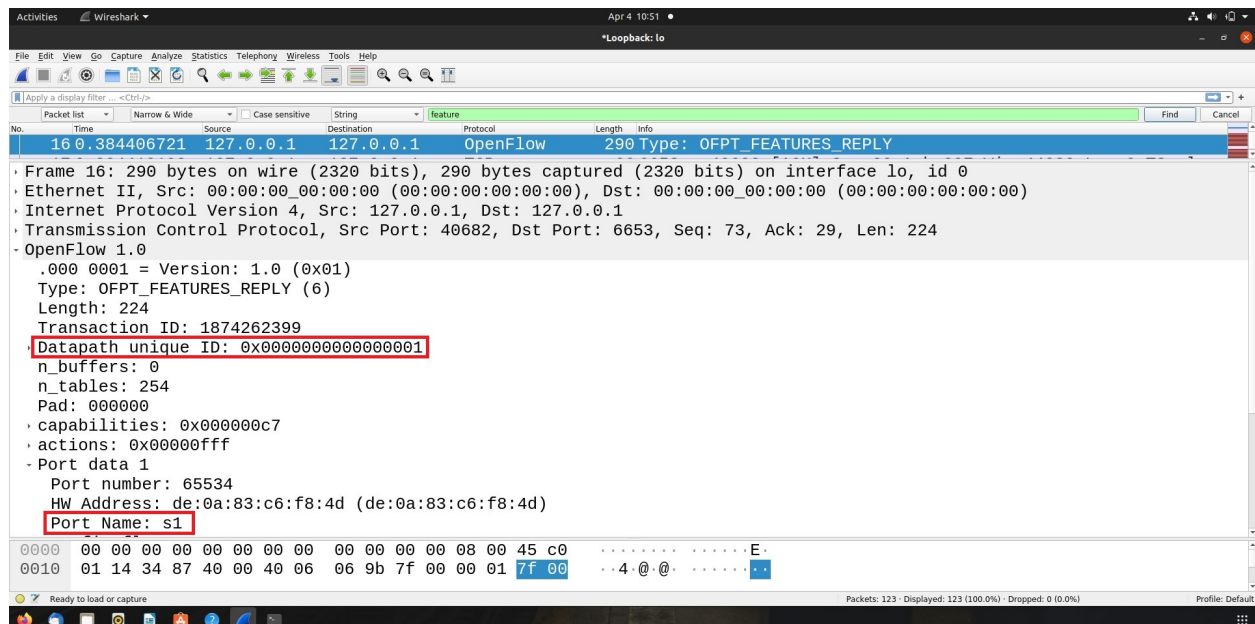
شکل ۳: پیام‌های *Feature request* و *Feature reply*

پیام *Feature request* توسط کنترلر به سوئیچ جهت گرفتن اطلاعات از قبیل *datapath ID* ارسال می‌شود.



شکل ۴: پیام Feature request

همانطور که در تصویر زیر می‌بینیم، پیام Feature reply حاوی datapath ID یکتا و اطلاعات درباره‌ی پورت‌های سوئیچ شامل نام هر پورت می‌باشد. همچنین سایر اطلاعات مربوطه در این پیام موجود است. پس از دریافت Feature request توسط سوئیچ، اطلاعات خواسته شده را به کنترلر ارسال می‌کند.



شکل ۵: پیام Feature reply

۴.۱

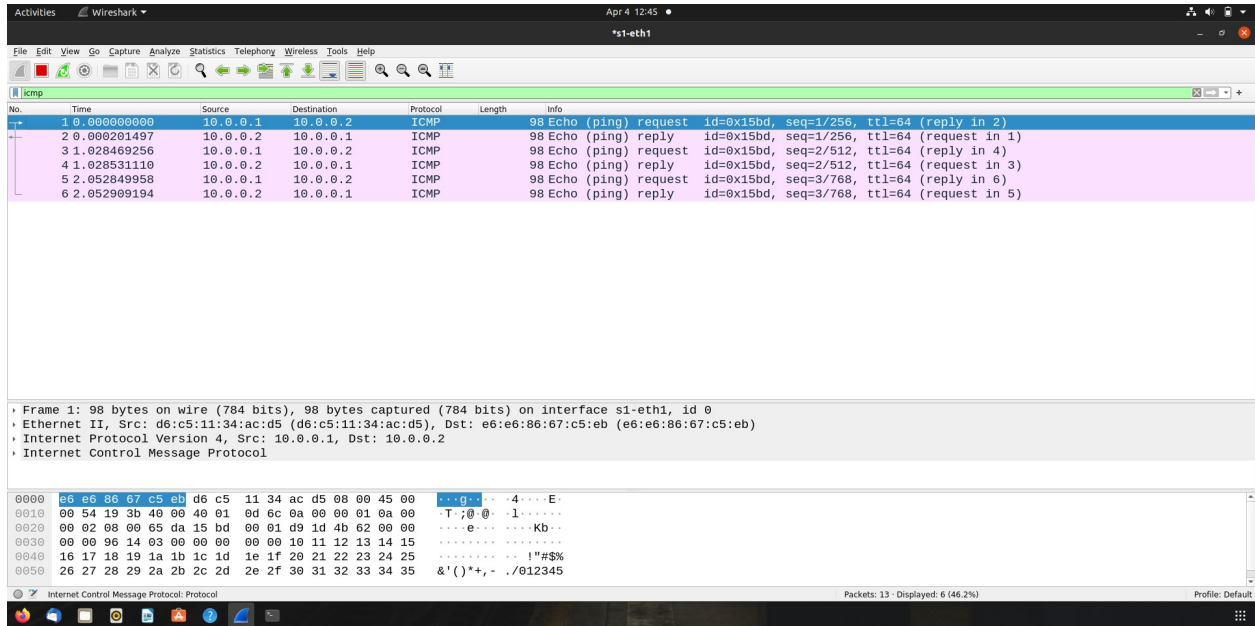
No.	Time	Source	Destination	Protocol	Length	Info
8	0.371857188	127.0.0.1	127.0.0.1	OpenFlow	74	Type: OFPT_HELLO
18	0.509914527	::	ff02::1:ff62:4073	OpenFlow	170	Type: OFPT_PACKET_IN
22	0.542465464	::	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
46	0.734322452	::	ff02::1:ffca:ca2c	OpenFlow	170	Type: OFPT_PACKET_IN
66	0.765680438	::	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
73	0.957522172	::	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
78	1.310354166	fe80::e41d:e3ff:fe40:5469	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
79	1.310488997	fe80::e41d:e3ff:fe40:5469	ff02::2	OpenFlow	154	Type: OFPT_PACKET_IN
84	1.397818412	fe80::e41d:e3ff:fe40:5469	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
88	1.534953591	fe80::4c57:efff:fe02:4073	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
89	1.534106313	fe80::4c57:efff:fe02:4073	ff02::2	OpenFlow	154	Type: OFPT_PACKET_IN
94	1.549832708	fe80::4c57:efff:fe02:4073	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
101	1.758088825	fe80::fcff:44ff:feca:ca2c	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
102	1.758226692	fe80::fcff:44ff:feca:ca2c	ff02::2	OpenFlow	154	Type: OFPT_PACKET_IN
108	2.461832451	fe80::fcff:44ff:feca:ca2c	ff02::16	OpenFlow	174	Type: OFPT_PACKET_IN
115	5.693828736	fe80::e41d:e3ff:fe40:5469	ff02::2	OpenFlow	154	Type: OFPT_PACKET_IN
118	5.949666879	fe80::4c57:efff:fe02:4073	ff02::2	OpenFlow	154	Type: OFPT_PACKET_IN
121	6.205901317	fe80::fcff:44ff:feca:ca2c	ff02::1:ff02:AB73	OpenFlow	174	Type: OFPT_PACKET_IN

شکل ۶: پیغام‌های Packet_in

۵.۱

به ازای همه پکت‌هایی که یا matching flow entry ندارند یا یک پکت با entry با match send to control action شود، یک پیغام packet_in برای کنترلر ارسال می‌شود.

۶.۱



No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.0.1	10.0.0.2	ICMP	98	Echo (ping) request id=0x15bd, seq=1/256, ttl=64 (reply in 2)
2	0.000201497	10.0.0.2	10.0.0.1	ICMP	98	Echo (ping) reply id=0x15bd, seq=1/256, ttl=64 (request in 1)
3	1.028469256	10.0.0.1	10.0.0.2	ICMP	98	Echo (ping) request id=0x15bd, seq=2/512, ttl=64 (reply in 4)
4	1.028531110	10.0.0.2	10.0.0.1	ICMP	98	Echo (ping) reply id=0x15bd, seq=2/512, ttl=64 (request in 3)
5	2.052849958	10.0.0.1	10.0.0.2	ICMP	98	Echo (ping) request id=0x15bd, seq=3/768, ttl=64 (reply in 6)
6	2.052909194	10.0.0.2	10.0.0.1	ICMP	98	Echo (ping) reply id=0x15bd, seq=3/768, ttl=64 (request in 5)

Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface s1-eth1, id 0
 Ethernet II, Src: d6:c5:11:34:ac:d5 (d6:c5:11:34:ac:d5), Dst: e6:e6:86:67:c5:eb (e6:e6:86:67:c5:eb)
 Internet Protocol Version 4, Src: 10.0.0.1, Dst: 10.0.0.2
 Internet Control Message Protocol

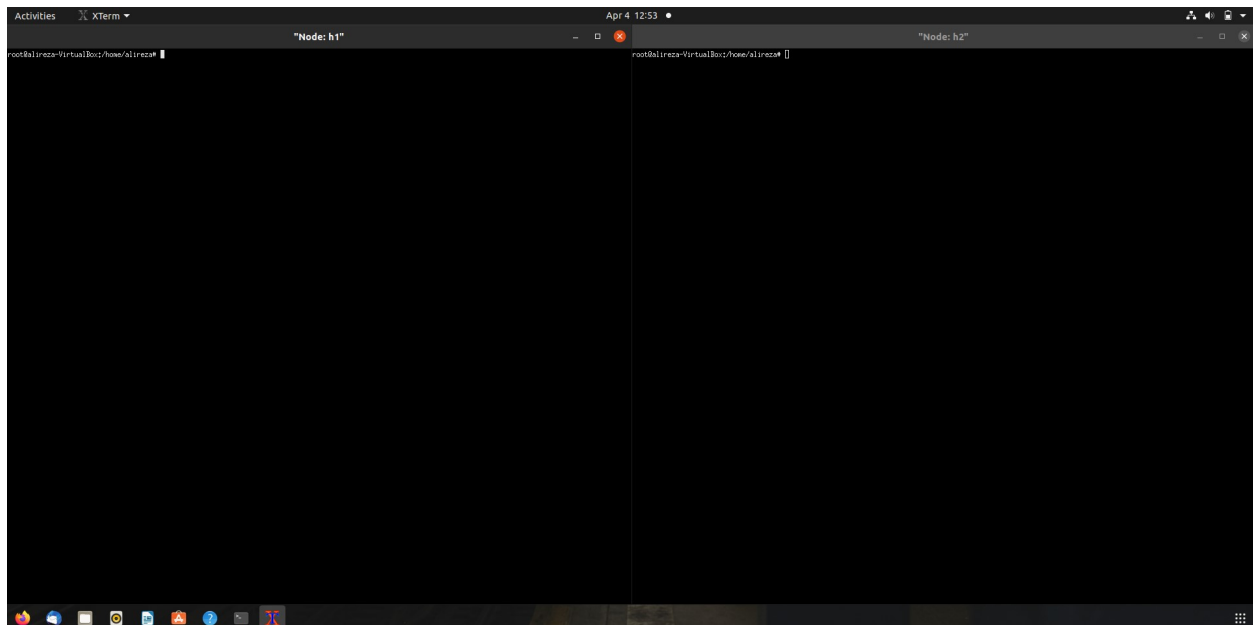
0000 e6 e6 86 67 c5 eb d6 c5 11 34 ac d5 08 00 45 00 ...g...-4---E:
 0010 00 54 19 3b 40 00 40 01 0d 6c 0a 00 00 01 0a 00 -T;@-0-1-----
 0020 00 02 08 00 65 da 15 bd 00 01 d9 1d 4b 62 00 00 ---e---Kb---
 0030 00 00 96 14 03 00 00 00 00 00 10 11 12 13 14 15
 0040 16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25!#\$%
 0050 26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35 &'()*+,-./012345

شکل ۷: $h1$ و $h2$ را پینگ می‌کند.

برای بسته‌های کیچر شده در وایرشارک از پروتکل ICMP استفاده شده است. از آنجایی که پینگ با ۳ پکت انجام شده است (هاست $h1$ پکت را برای $h2$ ارسال می‌کند)، وایرشارک ۳ پکت برای request و ۳ پکت برای reply را نمایش می‌دهد. در واقع به ازای هر پکتی request که $h1$ ارسال می‌کند، دریافت پیغام reply آن به معنی برقرار بودن ارتباط بین دو هاست است.

```
alireza@alireza-VirtualBox:~$ sudo mn --topo single,2
[sudo] password for alireza:
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1)
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet> xterm h1 h2
mininet>
```

شکل ۸: یک توپولوژی ساده با دو هاست را ایجاد می‌کنیم.



شکل ۹: با دستور *xterm* دو ترمینال جدا برای دو هاست *h1* و *h2* باز می‌کنیم.

شکل ۱۰: با دستور بالا یک سرور *TCP* در *h1* روی پورت ۵۰۰۰ به کار می‌اندازیم. همچنین در هر ثانیه نتایج مانیتور می‌شوند.

Interval	Transfer	Bandwidth
0.0- 1.0 sec	1.58 GBytes	13.6 Gbits/sec
1.0- 2.0 sec	1.73 GBytes	14.9 Gbits/sec
2.0- 3.0 sec	1.78 GBytes	15.3 Gbits/sec
3.0- 4.0 sec	1.78 GBytes	15.3 Gbits/sec
4.0- 5.0 sec	1.81 GBytes	15.5 Gbits/sec
5.0- 6.0 sec	1.77 GBytes	15.2 Gbits/sec
6.0- 7.0 sec	1.77 GBytes	15.2 Gbits/sec
7.0- 8.0 sec	1.78 GBytes	15.3 Gbits/sec
8.0- 9.0 sec	1.76 GBytes	15.2 Gbits/sec
9.0-10.0 sec	1.95 GBytes	16.8 Gbits/sec
10.0-11.0 sec	2.01 GBytes	17.3 Gbits/sec
11.0-12.0 sec	2.31 GBytes	19.8 Gbits/sec
12.0-13.0 sec	2.00 GBytes	17.1 Gbits/sec
13.0-14.0 sec	2.01 GBytes	17.2 Gbits/sec
14.0-15.0 sec	2.20 GBytes	18.9 Gbits/sec
15.0-16.0 sec	2.02 GBytes	17.3 Gbits/sec
16.0-17.0 sec	2.19 GBytes	18.9 Gbits/sec
17.0-18.0 sec	2.33 GBytes	20.0 Gbits/sec
18.0-19.0 sec	2.52 GBytes	21.6 Gbits/sec
19.0-20.0 sec	2.39 GBytes	20.5 Gbits/sec
0.0-20.0 sec	39.7 GBytes	17.0 Gbits/sec

شکل ۱۱: با دستور بالا یک کلاینت *TCP* در *h2* آغاز به کار می‌کند به سرور *TCP* (که آدرس آن با دستور *ifconfig* به دست می‌آید) متصل می‌شود. همچنین مدت زمان انتقال هم روی ۲۰ ثانیه با آپشن *-t* تنظیم شده است. همانطور که در تصویر می‌بینیم، از ۰ تا ۲۰ ثانیه، *throughput* میانگین برابر *17.0 Gbits/sec* است.

همچنین برای کشیدن نمودار می‌توانیم نتایج دستورات را درون فایل ذخیره کنیم و با استفاده از *gnuplot* نمودار بکشیم. در نهایت برای تست شبکه روی حالت *UDP*، از آپشن *-u* استفاده می‌کنیم.

منابع

- [1] <https://www.brianlinkletter.com/2015/09/how-to-map-openflow-switch-tcp-ports-in-mininet-sdn-simulations/>
- [2] <https://neelshelar.com/openflow-protocol-importance-message-types/>
- [3] http://csie.nqu.edu.tw/smallko/sdn/iperf_mininet.htm