پروژه دوم

مجتبی ملائی
۴۰۱۳۱۳۸۳

۱ تحلیل ترافیک شبکه

در این سوال از توپولوژی signel با ۳ میزبان استفاده شد و سویچ به یک کنترل خارجی با پروتکل OpenFlow۱.۳ متصل گشت. بسته های میزبان اول و همچنین OpenFlow ذخیره شده اند. پس از اجرا، میزبان h1 از میزبان h2 پینگ می‌گیرد.
در میزبان اول بسته های زیر دیده می‌شوند:

- ICMPv6 Router Solicitation: بسته هایی هستند که میزبان ها برای پیدا کردن روتر در پروتکل IPv6 ارسال می‌کنند. (در این سوال کاربردی ندارند.)
- ARP: برای پیدا کردن mac-address متناظر با یک IP در یک شبکه محلی
- ICMP: برای پینگ گرفتن

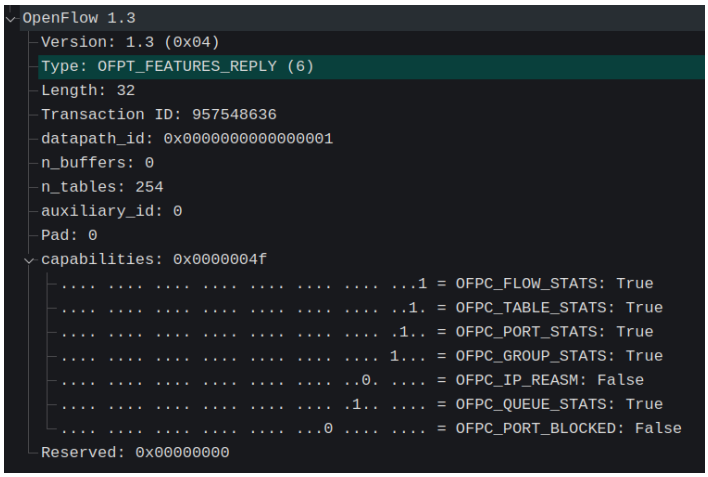
در کنترل خارجی بسته های زیر دیده می‌شود:

- TCP: ارتباط قابل اطمینان بین کنترلر و سوئیچ
- OpenFlow: ارتباط و انتقال دستورات و اطلاعات میان کنترلر و سوئیچ

آ) در پروتکل OpenFlow، هدف از تبادل پیام‌های Feature Request و Feature Reply بین سوئیچ و کنترلر، شناسایی قابلیت‌ها و ویژگی‌های سوئیچ توسط کنترلر است. این تبادل به کنترلر اجازه می‌دهد تا از امکانات سخت‌افزاری و نرم‌افزاری سوئیچ آگاه شود و بر اساس آن، تصمیم‌گیری‌های مناسب انجام دهد.برخی از این اطلاعات شامل:

- شناسه سوئیچ (Datapath ID)
- تعداد جدول‌ها (n_tables): تعداد جدول‌هایی که سوئیچ پشتیبانی می‌کند
- قابلیت‌ها

No.	Time	Source	Destination	Protocol	Lengtl	Info
14	0.162225	127.0.0.1	127.0.0.1	OpenFlow	74	Type: OFPT_FEATURES_REQUEST
16	0.164687	127.0.0.1	127.0.0.1	OpenFlow	98	Type: OFPT_FEATURES_REPLY

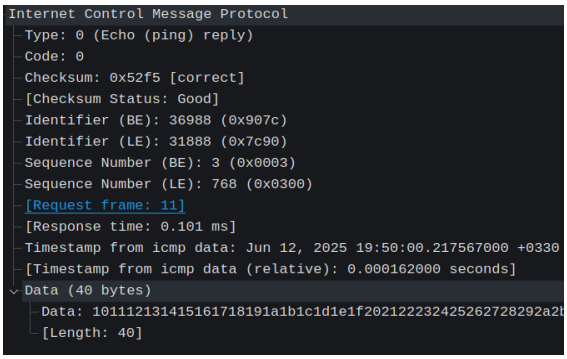


ب) در این دو سناریو بسته به کنترلر ارسال می‌شود.

- `OFPR_NO_MATCH(0)`: وقتی هیچ Flow ای برای بسته در جدول وجود نداشته باشد و اصطلاحا miss رخ دهد.
- `OFPR_ACTION(1)`: وقتی در Flow متناظر با آن بسته action آن ارسال به کنترلر باشد.

ج) در ارسال بسته های ICMP از پروتکل IP استفاده می‌شود. ابتدا با استفاده از ARP به mac-address مورد نظر دست پیدا می‌کند. سپس بسته های ICMP در بسته های IP و سپس در فریم Ethernet قرار می‌گیرند. در این پروتکل مقدار type در بسته ها ICMP برابر با ۸ قرار می‌گیرد که به معنی Echo Request است. و در پاسخ این فیلد مقدار 0 می‌گیرد که به معنای Echo Reply است. از فیلد ident برای متمایز کردن پینگ بین پروسس های متفاوت استفاده می‌شود و از seq برای ترتیب و متمایز کردن بسته ها استفاده می‌شود. همچنین هر بسته دارای timestamp می‌باشد.

7	14.100371	10.0.0.1	10.0.0.2	ICMP	98	Echo (ping) request id=0x907c, seq=1/256, ttl=64 (reply in 8)
8	14.102988	10.0.0.2	10.0.0.1	ICMP	98	Echo (ping) reply id=0x907c, seq=1/256, ttl=64 (request in 7)
9	15.096396	10.0.0.1	10.0.0.2	ICMP	98	Echo (ping) request id=0x907c, seq=2/512, ttl=64 (reply in 10)
10	15.097049	10.0.0.2	10.0.0.1	ICMP	98	Echo (ping) reply id=0x907c, seq=2/512, ttl=64 (request in 9)
11	16.123638	10.0.0.1	10.0.0.2	ICMP	98	Echo (ping) request id=0x907c, seq=3/768, ttl=64 (reply in 12)
12	16.123739	10.0.0.2	10.0.0.1	ICMP	98	Echo (ping) reply id=0x907c, seq=3/768, ttl=64 (request in 11)
13	17.147766	10.0.0.1	10.0.0.2	ICMP	98	Echo (ping) request id=0x907c, seq=4/1024, ttl=64 (reply in 14)
14	17.147881	10.0.0.2	10.0.0.1	ICMP	98	Echo (ping) reply id=0x907c, seq=4/1024, ttl=64 (request in 13)
15	18.171783	10.0.0.1	10.0.0.2	ICMP	98	Echo (ping) request id=0x907c, seq=5/1280, ttl=64 (reply in 16)
16	18.171895	10.0.0.2	10.0.0.1	ICMP	98	Echo (ping) reply id=0x907c, seq=5/1280, ttl=64 (request in 15)



۲ اجرای توپولوژی‌های مختلف در Mininet

۱. `sudo mn --topo single,5 --mac --switch ovsk` : این دستور از توپولوژی سینگل (یک سوئیچ مرکزی) و ۵ میزبان که به آن متصل هستند استفاده می‌کند. mac-address ها به صورت اتوماتیک تنظیم می‌شوند و از سویچ مجازی در کرنل برای سویچ استفاده می‌شود.

۲. `sudo mn --topo single,5 --controller remote -x` : این دستور از توپولوژی سینگل (یک سوئیچ مرکزی) و ۵ میزبان که به آن متصل هستند استفاده می‌کند. آپشن کنترلر باعث می‌شود تا سویچ مرکزی به یک کنترلر خارجی SDN بر روی پورت ۶۶۳۳ یا ۶۶۵۳ طبق پروتکل OpenFlow متصل شود. اپشن x نیز به ازای هر دستگاه (میزبان، سوئیچ، کنترلر) یک shell ایجاد می‌کند.

۳. `sudo mn --topo tree,5 --mac --arp` : این دستور از توپولوژی درخت به شکل باینری و عمق ۵ (فقط سوئیچ) و میزبان ها که به برگ ها وصل شده اند استفاده می‌کند. ۳۲ میزبان و ۳۱ سوئیچ. mac-address ها به صورت اتوماتیک تنظیم می‌شوند. همچنین مقدار های arp-entry کل شبکه برای تمام میزبان ها از قبل تنظیم می‌شود.

۴. `sudo mn --topo linear --controller=remote,ip=127.1.0.0,port=6633` : این دستور از توپولوژی خطی (دو میزبان و دو سوئیچ) استفاده می‌کند. آپشن کنترلر باعث می‌شود تا سوئیچ ها به یک کنترلر خارجی SDN بر روی آدرس IP و پورت مشخص شده متصل شوند.

۱. به ترتیب: سینگل، سینگل، درختی و خطی

۲.
- `--mac` : این دستور mac-address ها را برای میزبان ها به صورت مشخص از `00:00:00:00:00:01` شروع به تخصیص می‌کند. مثلا mac-address میزبان h10 برابر خواهد بود با `00:00:00:00:00:10` این کار دیباگ کردن شبکه را آسانتر می‌کند.
- `--arp` : این آپشن arp-entity ها را به صورت خودکار برای همه میزبان ها پر می‌کند. بنابراین نیازی به اجرای پروتکل arp نیست.

۳

۱. از همه میزبان ها به یکدیگر پینگ گرفته شد و نشان میدهد شبکه به درستی کار می‌کند.

```
sudo python tree_topo.py 2
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2 h3 h4
*** Adding switches:
s1 s2 s3
*** Adding links:
(s1, s2) (s1, s3) (s2, h1) (s2, h2) (s3, h3) (s3, h4)
*** Configuring hosts
h1 h2 h3 h4
*** Starting controller
c0
*** Starting 3 switches
s1 s2 s3 ...
*** Starting CLI:
mininet> h1 ping h2
PING 10.0.0.2 (10.0.0.2) 56(84) bytes of data.
64 bytes from 10.0.0.2: icmp_seq=1 ttl=64 time=3.58 ms
64 bytes from 10.0.0.2: icmp_seq=2 ttl=64 time=0.997 ms
^C
--- 10.0.0.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 0.997/2.288/3.580/1.291 ms
mininet> h1 ping h3
PING 10.0.0.3 (10.0.0.3) 56(84) bytes of data.
64 bytes from 10.0.0.3: icmp_seq=1 ttl=64 time=5.76 ms
64 bytes from 10.0.0.3: icmp_seq=2 ttl=64 time=2.11 ms
^C
--- 10.0.0.3 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 2.105/3.931/5.758/1.826 ms
mininet> h1 ping h4
PING 10.0.0.4 (10.0.0.4) 56(84) bytes of data.
64 bytes from 10.0.0.4: icmp_seq=1 ttl=64 time=5.76 ms
64 bytes from 10.0.0.4: icmp_seq=2 ttl=64 time=2.06 ms
64 bytes from 10.0.0.4: icmp_seq=3 ttl=64 time=0.171 ms
^C
--- 10.0.0.4 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 0.171/2.666/5.764/2.322 ms
mininet> h2 ping h3
PING 10.0.0.3 (10.0.0.3) 56(84) bytes of data.
64 bytes from 10.0.0.3: icmp_seq=1 ttl=64 time=5.84 ms
64 bytes from 10.0.0.3: icmp_seq=2 ttl=64 time=2.18 ms
64 bytes from 10.0.0.3: icmp_seq=3 ttl=64 time=0.094 ms
^C
--- 10.0.0.3 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 0.094/2.705/5.837/2.373 ms
mininet> h2 ping h4
PING 10.0.0.4 (10.0.0.4) 56(84) bytes of data.
64 bytes from 10.0.0.4: icmp_seq=1 ttl=64 time=5.10 ms
64 bytes from 10.0.0.4: icmp_seq=2 ttl=64 time=1.66 ms
^C
--- 10.0.0.4 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.661/3.381/5.101/1.720 ms
mininet> h3 ping h4
PING 10.0.0.4 (10.0.0.4) 56(84) bytes of data.
64 bytes from 10.0.0.4: icmp_seq=1 ttl=64 time=2.40 ms
64 bytes from 10.0.0.4: icmp_seq=2 ttl=64 time=1.14 ms
^C
--- 10.0.0.4 ping statistics ---
```

```
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 1.136/1.766/2.396/0.630 ms
mininet>
```

۲. • `pingall` : از هر میزبان به میزبان های دیگر پینگ می‌گیرد.

```
mininet> pingall
*** Ping: testing ping reachability
h1 -> h2 h3 h4
h2 -> h1 h3 h4
h3 -> h1 h2 h4
h4 -> h1 h2 h3
*** Results: 0% dropped (12/12 received)
```

• `nodes` : اجزای شبکه را نشان میدهد.

```
mininet> nodes
available nodes are:
c0 h1 h2 h3 h4 s1 s2 s3
```

- `c0` : کنترلر خارجی
- `sx` : سوئیچ
- `hx` : میزبان

۳. `dump` : این دستور اطلاعات دقیق و کاملی در مورد هر نود شبکه به ما میدهد. از جمله آن می توان به نام، نوع، اینترفیس، آدرسIP ، شماره پروسس اشاره کرد. در این خروجی سوئیچ به دلیل آنکه آدرس آیپی ندارد، به ازای همه اینترفیس ها بجز None lo آمده است. همچنین نوع سوئیچ OVSSwitch می باشد.

```
mininet> dump
<Host h1: h1-eth0:10.0.0.1 pid=5400>
<Host h2: h2-eth0:10.0.0.2 pid=5402>
<Host h3: h3-eth0:10.0.0.3 pid=5404>
<Host h4: h4-eth0:10.0.0.4 pid=5406>
<OVSSwitch s1: lo:127.0.0.1,s1-eth1:None,s1-eth2:None pid=5411>
<OVSSwitch s2: lo:127.0.0.1,s2-eth1:None,s2-eth2:None,s2-eth3:None pid=5414>
<OVSSwitch s3: lo:127.0.0.1,s3-eth1:None,s3-eth2:None,s3-eth3:None pid=5417>
<Controller c0: 127.0.0.1:6653 pid=5393>
```