

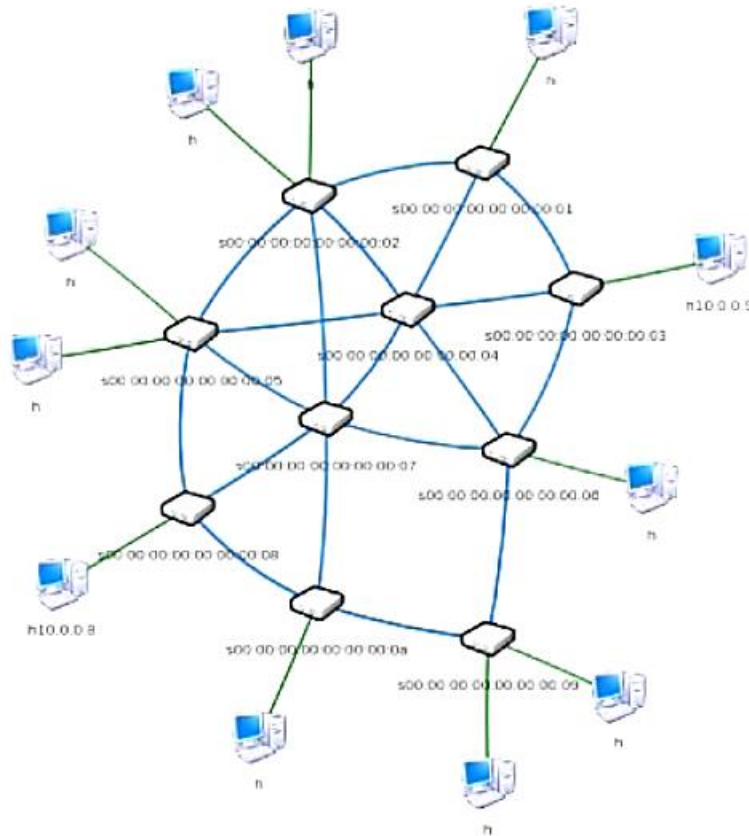
سوال ۱: انجام شد

سوال ۲: کد توپولوژی موردنظر شامل هاست ها و سویچ ها و لینک های بین آنها نوشته شده و فایل آن به نام topo_Q۲.py ضمیمه شده است.

با استفاده از دستور زیر آن را میسازیم و فلودلایت را به عنوان کنترلر آن استفاده میکنیم:

`Sudo mn -custom topo_Q2.py -topo mytopo -controller=remote,ip=127.0.0.1 , port=6653`

تصویر توپولوژی:



۱-۲- حال با دستور زیر از هاست ۹ به هاست ۸ پینگ میکنیم:

Mininet> h9 ping h8

و میبینیم که پینگ با موفقیت انجام میشود.

۱-۱-۲- حال جداول سویچ های در مسیر را بررسی میکنیم تا ببینیم بسته ها از چه مسیری به مقصد میرسند طبق بررسی ها و توضیحاتی که کامل در ویدیو ضمیمه شده است، میبینیم که مسیر زیر انتخاب شده است (۳) عدد کوتاه ترین مسیر بین هاست ۹ و ۸ در این توپولوژی وجود دارد و مسیر زیر نیز یکی از آنهاست)

h9→s3→s4→s5→s8→h8

همانطور که در ویدیو توضیح داده شده چون به طور دستی داده ای وارد جداول نکرده ایم در این مرحله الگوریتم های مسیریابی کوتاهترین مسیر را انتخاب میکنند. شایان ذکر است که جدول تمام سویچ های در مسیر در ویدیو وجود داشته و توضیحات مهم در خصوص مسیر انتخاب شده در ویدیو داده شده است.

۲-۲- در اینجا با استفاده از کدی که ضمیمه شده (flow_Q2.py) با وارد کردن flow های لازم به جداول با استفاده از دستور زیر، سویچ ها را به گونه ای تنظیم میکنیم که ترافیک بین هاست ۹ به هاست ۸ از مسیر مورد نظر ما عبور کند:

Python flow_Q2.py

این امر به این شکل است که با توجه به مبدا و قصد هر بسته و پورت ورودی آن به هر سویچ اکشنی نظیر آن انتخاب شده است (اینکه بسته از کدام پورت خارج شود) تا مطمئن شویم بسته ها از مسیر مورد نظر ما عبور میکنند:

h9→s3→s1→s4→s6→s7→s8→h8

۲-۳- حال با دستور زیر از هاست ۹ به هاست ۸ پینگ میکنیم:

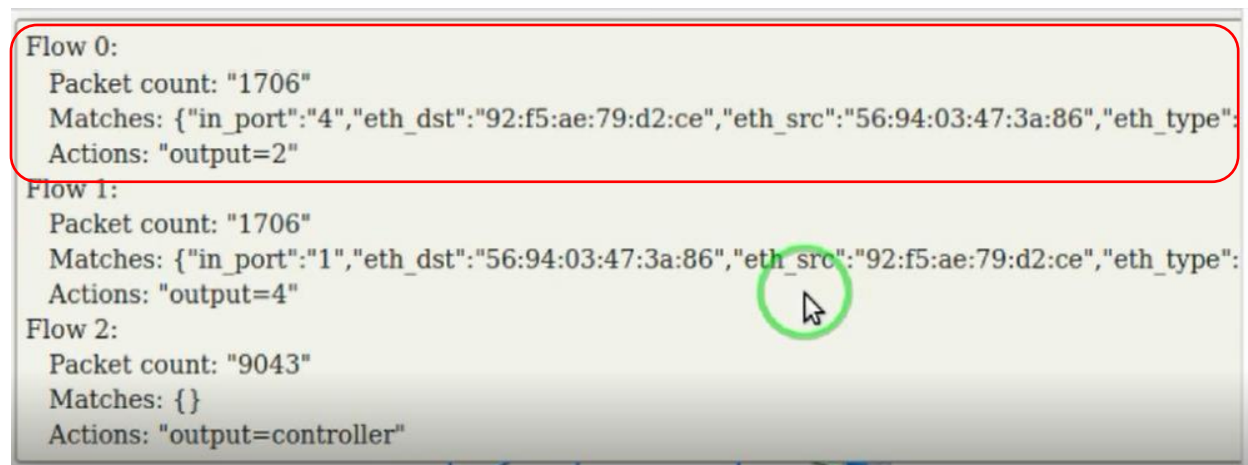
Mininet> h9 ping h8

و میبینیم که پینگ با موفقیت انجام میشود.

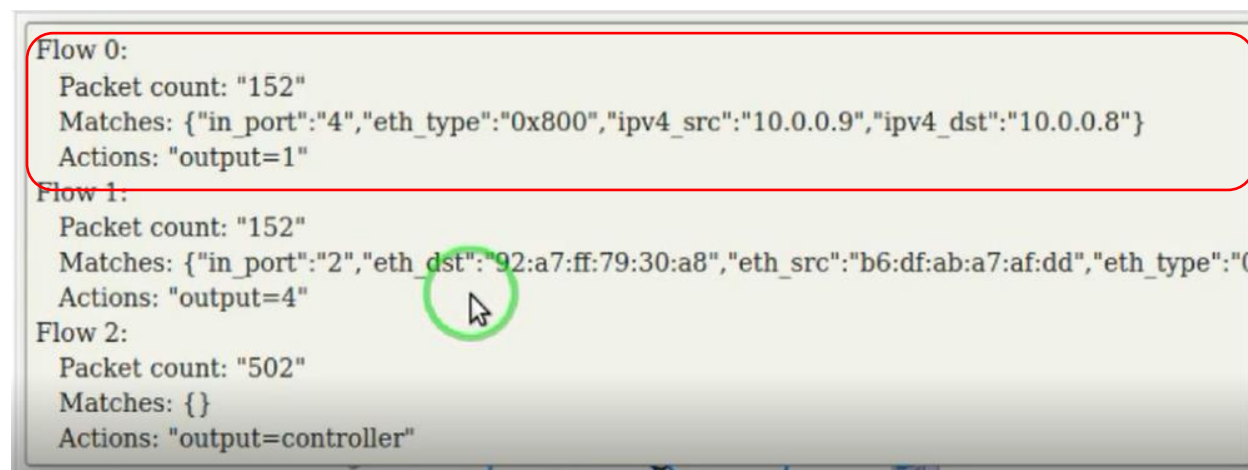
۱-۳-۲- در این قسمت دو سویچ را مورد بررسی قرار میدهیم و میبینیم که جداول آنها نسبت به حالت قبل چه تغییری داشته اند :

Switch 3 :

در ابتدا میبینیم طی مسیریابی انجام شده بسته های (پینگ با مبدا میزبان ۹ و مقصد میزبان ۸) که از پورت ۴ به سویچ ۳ وارد میشوند به طرف پورت ۲ هدایت میشوند این امر به این دلیل است که در کوتاهترین مسیر تشخیص داده شده (h9→s3→s4→s5→s8→h8)، بسته ای که از مبدا میزبان ۹ به سویچ ۳ آمده و میخواهد به میزبان ۸ برود باید به سمت سویچ ۴ هدایت شود که به این منظور در این سویچ باید از پورت خروجی ۲ در استفاده کند.



پس از اعمال تغییرات در جدول سویچ ها با کد مورد نظر، میبینیم که بسته های مورد نظر (پینگ با مبدا میزبان ۹ و مقصد میزبان ۸) که از پورت ۴ به سویچ ۳ وارد میشوند این بار، به طرف پورت ۱ هدایت میشوند این امر به این دلیل است که طبق مسیر مورد نظری که ورودی های جدول را در کد طبق آن تنظیم کردیم (h9→s3→s1→s4→s6→s7→s8→h8)، بسته ای که از مبدا میزبان ۹ به سویچ ۳ آمده و میخواهد به میزبان ۸ برود باید به سمت سویچ ۱ هدایت شود که به این منظور در این سویچ باید از پورت خروجی ۱ استفاده کند.



Switch 4 :

در ابتدا میبینیم طی مسیریابی انجام شده بسته های (پینگ با مبدا میزبان ۹ و مقصد میزبان ۸) که از پورت ۲ به سویچ ۴ وارد میشوند به طرف پورت ۳ هدایت میشوند این امر به این دلیل است که در کوتاهترین مسیر تشخیص داده شده (h9→s3→s4→s5→s8→h8)، بسته ای که از مبدا میزبان ۹ به سویچ ۴ آمده و میخواهد به میزبان ۸ برود باید به سمت سویچ ۵ هدایت شود که به این منظور در این سویچ باید از پورت خروجی ۳ در استفاده کند.

Flow 0:
Packet count: "1735"
Matches: {"in_port": "2", "eth_dst": "92:f5:ae:79:d2:ce", "eth_src": "56:94:03:47:3a:86", "eth_type": "0800"}
Actions: "output=3"
Flow 1:
Packet count: "1735"
Matches: {"in_port": "3", "eth_dst": "56:94:03:47:3a:86", "eth_src": "92:f5:ae:79:d2:ce", "eth_type": "0800"}
Actions: "output=1"
Flow 2:
Packet count: "9436"
Matches: {}
Actions: "output=controller"

پس از اعمال تغییرات در جدول سویچ ها با کد موردنظر، میبینیم که بسته های مورد نظر (پینگ با مبدا میزبان ۹ و مقصد میزبان ۸) که از پورت ۱ به سویچ ۴ وارد میشوند این بار، به طرف پورت ۶ هدایت میشوند این امر به این دلیل است که طبق مسیر مورد نظری که ورودی های جدول را در کد طبق آن تنظیم کردیم (h9→s3→s1→s4→s6→s7→s8→h8)، بسته ای که از مبدا میزبان ۹ به سویچ ۴ آمده و میخواهد به میزبان ۸ برود باید به سمت سویچ ۱۶ هدایت شود که به این منظور در این سویچ باید از پورت خروجی ۶ استفاده کند. در این مورد تفاوت دیگری نیز قابل توجه است و آن اینکه بسته های با مبدا هاست ۹ و مقصد هاست ۸ نسبت به حالت قبل پورت ورودی متفاوتی نیز دارند که این به همان دلیل تغییر مسیر بسته ها به مسیر گفته شده است. در حالت کوتاه ترین مسیر بسته ها از سویچ ۳ به سویچ ۴ میآمدند که از طریق پورت ورودی ۲ بود اما در حالت مسیری که خودمان وارد کردیم از سویچ ۱ به سویچ ۴ میآیند که از طریق پورت ورودی ۱ است.

Flow 0:
Packet count: "154"
Matches: {"in_port": "1", "eth_type": "0x800", "ipv4_src": "10.0.0.9", "ipv4_dst": "10.0.0.8"}
Actions: "output=6"
Flow 1:
Packet count: "154"
Matches: {"in_port": "3", "eth_dst": "92:a7:ff:79:30:a8", "eth_src": "b6:df:ab:a7:af:dd", "eth_type": "0800"}
Actions: "output=2"
Flow 2:
Packet count: "635"
Matches: {}
Actions: "output=controller"

۴-۲- همانطور که از جداول مشخص است و طبق اطلاعاتی که خودمان وارد جدول کردیم این بار مسیر زیر را خواهیم داشت:

$h9 \rightarrow s3 \rightarrow s1 \rightarrow s4 \rightarrow s6 \rightarrow s7 \rightarrow s8 \rightarrow h8$

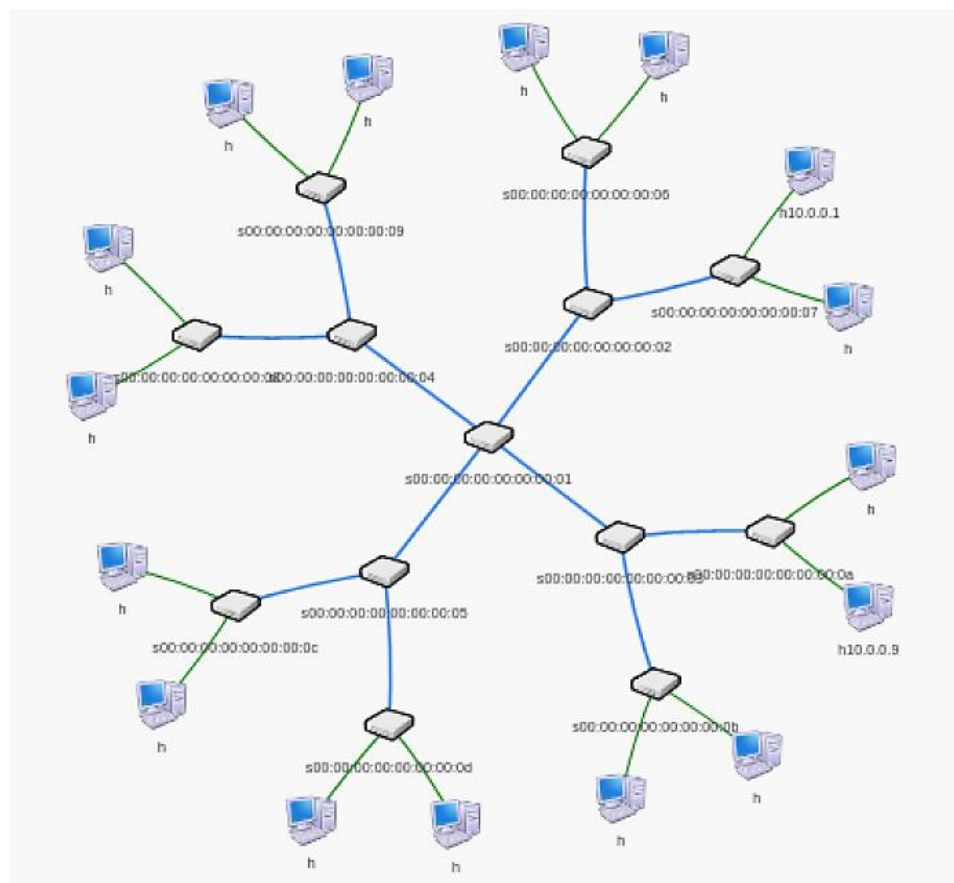
بررسی دقیق این موضوع با توجه به مقادیر موجود در جداول در ویدیو توضیح داده شده است. به طور خلاصه میتوان گفت با بررسی فیلدهایی که مبدأ آنها هاست ۹ و مقصدشان هاست ۸ است و با توجه به پورت ورودی آنها به هر سویچ، اکشن مورد نظر در آن سویچ بررسی شده و مشخص شد بسته ها از مسیر گفته شده عبور میکنند (بر عکس حالت دیفالت که از کوتاه ترین مسیر عبور میکردند). توضیحات بیشتر و مقادیر دقیق جداول در مسیر، در ویدیو قابل مشاهده است.

سوال ۳: کد مورد نظر برای ایجاد توپولوژی پروژه ۱ در فایل topo_Q3.py ضمیمه شده است.

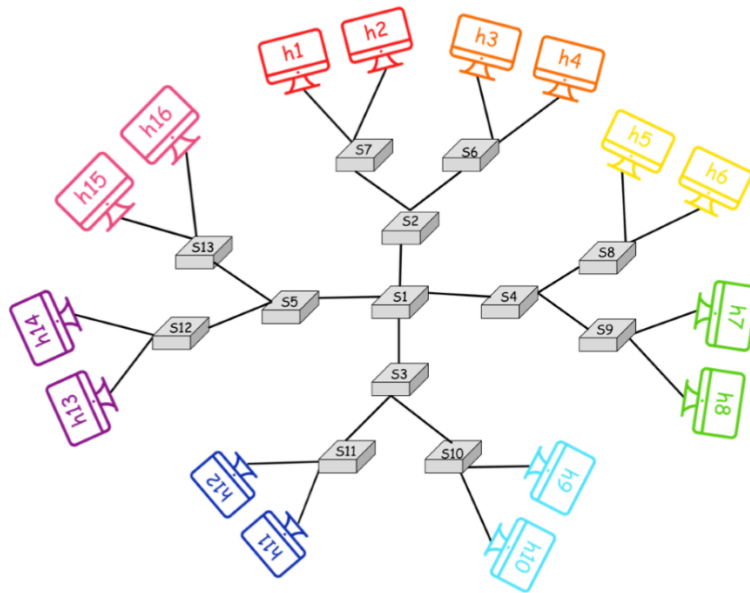
با استفاده از دستور زیر آن را میسازیم و فلودلایت را به عنوان کنترلر آن استفاده میکنیم:

```
Sudo mn -custom topo_Q3.py -topo mytopo2 -controller=remote,ip=127.0.0.1 , port=6653
```

شکل توپولوژی:



۱-۳- مسیر های نوشته شده در پروژه ۱ به صورت زیر بودند :



شش مسیر با حداکثر طول و حداقل تلاقی :

$h1 \rightarrow S7 \rightarrow S2 \rightarrow S1 \rightarrow S3 \rightarrow S10 \rightarrow h9$
 $h5 \rightarrow S8 \rightarrow S4 \rightarrow S1 \rightarrow S5 \rightarrow S13 \rightarrow h16$
 $h11 \rightarrow S11 \rightarrow S3 \rightarrow S1 \rightarrow S5 \rightarrow S12 \rightarrow h13$
 $h4 \rightarrow S6 \rightarrow S2 \rightarrow S1 \rightarrow S4 \rightarrow S9 \rightarrow h8$
 $h2 \rightarrow S7 \rightarrow S2 \rightarrow S1 \rightarrow S5 \rightarrow S13 \rightarrow h15$
 $H6 \rightarrow S8 \rightarrow S4 \rightarrow S1 \rightarrow S3 \rightarrow S10 \rightarrow h10$

با توجه به مبدا و قصد هر بسته و پورت ورودی آن به هر سویچ اکشنی نظیر آن انتخاب شده است (اینکه بسته از کدام پورت خارج شود) تا مطمئن شویم بسته ها از مسیر مورد نظر ما عبور میکنند.

کد flow_Q3.py به این منظور نوشته شده است که با دستور زیر قابل اجرا است:

Python flow_Q3.py

اینکه برای هر مسیر دقیقا چه مقداری به جداول اضافه میشود در کد قابل مشاهده است. همانطور که در صورت سوال خواسته شده یکی از این مسیرها (مسیر اول) نحوه ی کار سویچ ها و پیکر بندی دقیق آنها در ویدیو آورده شده است. لینک ویدیوها در صفحه ی بعد ضمیمه شده

لینک ویدیو ها :

<https://iutbox.iut.ac.ir/index.php/s/YknPwcQiiJbjtC۸>