باسمه تعالی

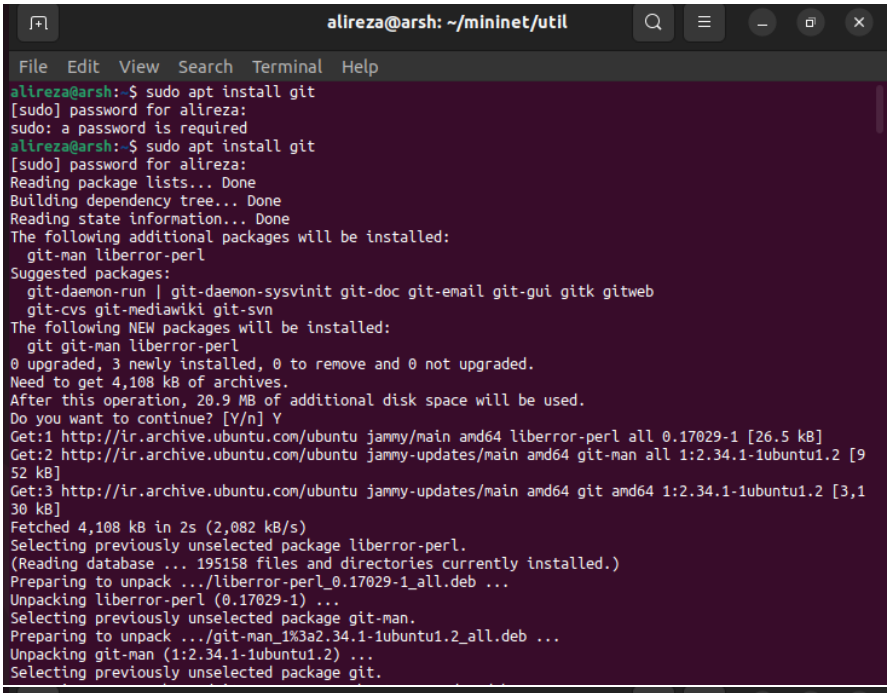
دانشگاه صنعتی اصفهان

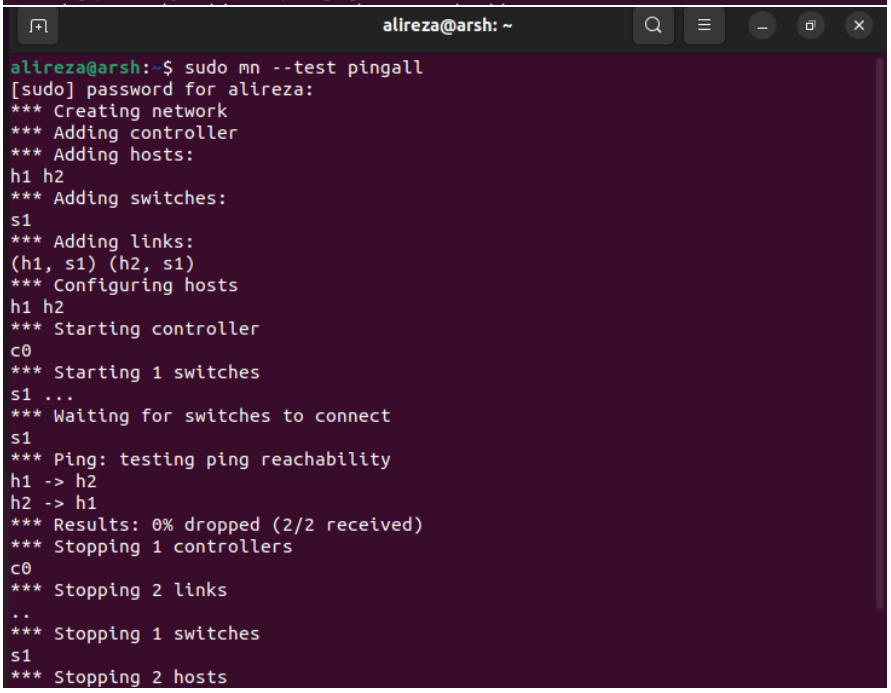
دانشکده مهندسی برق و کامپیوتر

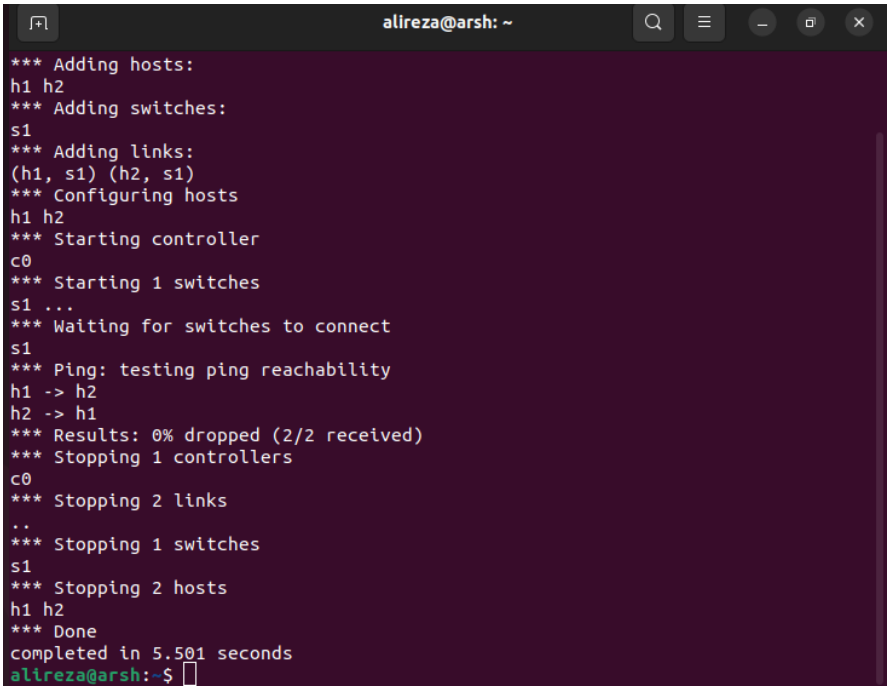
شبکه های کامپیوتری۲ – پروژه یک

علیرضا صالحی حسین آبادی - ۹۷۲۹۸۳۳

۱.


alireza@arsh: ~/mininet/util
File Edit View Search Terminal Help
alireza@arsh:~$ sudo apt install git
[sudo] password for alireza:
sudo: a password is required
alireza@arsh:~$ sudo apt install git
[sudo] password for alireza:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
git-man liberror-perl
Suggested packages:
git-daemon-run | git-daemon-sysvinit git-doc git-email git-gui gitk gitweb
git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
git git-man liberror-perl
0 upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 4,108 kB of archives.
After this operation, 20.9 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://ir.archive.ubuntu.com/ubuntu jammy/main amd64 liberror-perl all 0.17029-1 [26.5 kB]
Get:2 http://ir.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git-man all 1:2.34.1-1ubuntu1.2 [952 kB]
Get:3 http://ir.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git amd64 1:2.34.1-1ubuntu1.2 [3,130 kB]
Fetched 4,108 kB in 2s (2,082 kB/s)
Selecting previously unselected package liberror-perl.
(Reading database ... 195158 files and directories currently installed.)
Preparing to unpack .../liberror-perl_0.17029-1_all.deb ...
Unpacking liberror-perl (0.17029-1) ...
Selecting previously unselected package git-man.
Preparing to unpack .../git-man_1%3a2.34.1-1ubuntu1.2_all.deb ...
Unpacking git-man (1:2.34.1-1ubuntu1.2) ...
Selecting previously unselected package git.


alireza@arsh: ~
alireza@arsh:~$ sudo mn --test pingall
[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 ...
*** Waiting for switches to connect
s1
*** Ping: testing ping reachability
h1 -> h2
h2 -> h1
*** Results: 0% dropped (2/2 received)
*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts


alireza@arsh: ~
*** Adding hosts:
h1 h2
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1)
*** Configuring hosts
h1 h2
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Waiting for switches to connect
s1
*** Ping: testing ping reachability
h1 -> h2
h2 -> h1
*** Results: 0% dropped (2/2 received)
*** Stopping 1 controllers
c0
*** Stopping 2 links
..
*** Stopping 1 switches
s1
*** Stopping 2 hosts
h1 h2
*** Done
completed in 5.501 seconds
alireza@arsh:~$

۲.

- ساختن شبکه Single با سه میزبان

```
alireza@arsh:~$ sudo mn --topo single,3
[sudo] password for alireza:
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2 h3
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1) (h3, s1)
*** Configuring hosts
h1 h2 h3
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
```

- اجرای فرمان mininnet->nodes: نمایش نودهای در دسترس(available)

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

خروجی: یک کنترلر(c0)، سه میزبان(h1، h2 و h3)، یک سوئیچ(s1)

- اجرای فرمان mininet-> net: نمایش لینک‌های در دسترس(available)

```
mininet> net
h1 h1-eth0:s1-eth1
h2 h2-eth0:s1-eth2
h3 h3-eth0:s1-eth3
s1 lo:  s1-eth1:h1-eth0 s1-eth2:h2-eth0 s1-eth3:h3-eth0
c0
```

✓ خروجی:

❖ میزبان h1 با استفاده از رابط شبکه(network interface) h1-eth0 به سوئیچ روی رابط s1-eth1 متصل می شود.

میزبان h2 با استفاده از رابط شبکه h2-eth0 به سوئیچ روی رابط s1-eth2 متصل می شود.

❖ میزبان h3 با استفاده از رابط شبکه h3-eth0 به سوئیچ روی رابط s1-eth3 متصل می شود.

❖ سوئیچ s1:

- دارای یک رابط(interface) Loopback lo.
- از طریق رابط(interface) s1-eth1 به h1-eth0 متصل می شود.
- از طریق رابط(interface) s1-eth2 به h2-eth0 متصل می شود.
- از طریق رابط(interface) s1-eth3 به h3-eth0 متصل می شود.

❖ کنترلر c0 مغز شبکه است که در آن دانش جهانی در مورد شبکه دارد. یک کنترلر به سوئیچ ها دستور می دهد که چگونه بسته ها را در شبکه فوروارد یا رها کنند.

- اجرای دستور mininnet->dump: گرفتن اطلاعات در مورد همه گره‌ها

```
mininet> dump
<Host h1: h1-eth0:10.0.0.1 pid=50864>
<Host h2: h2-eth0:10.0.0.2 pid=50866>
<Host h3: h3-eth0:10.0.0.3 pid=50868>
<OVSSwitch s1: lo:127.0.0.1,s1-eth1:None,s1-eth2:None,s1-eth3:None pid=50873>
<Controller c0: 127.0.0.1:6653 pid=50857>
```

✓ خروجی:

❖ میزبان h1 دارای یک رابط h1-eth0 است که با آدرس IP = 10.0.0.1 پیکربندی شده است. همچنین h1_PID = 50864 است.

❖ میزبان h2 دارای یک رابط h2-eth0 است که با آدرس IP = 10.0.0.2 پیکربندی شده است. همچنین h2_PID = 50866 است.

میزبان h3 دارای یک رابط h3-eth0 است که با آدرس IP = 10.0.0.3 پیکربندی شده است. همچنین h3_PID = 50868 است.

❖ سوئیچ s1:

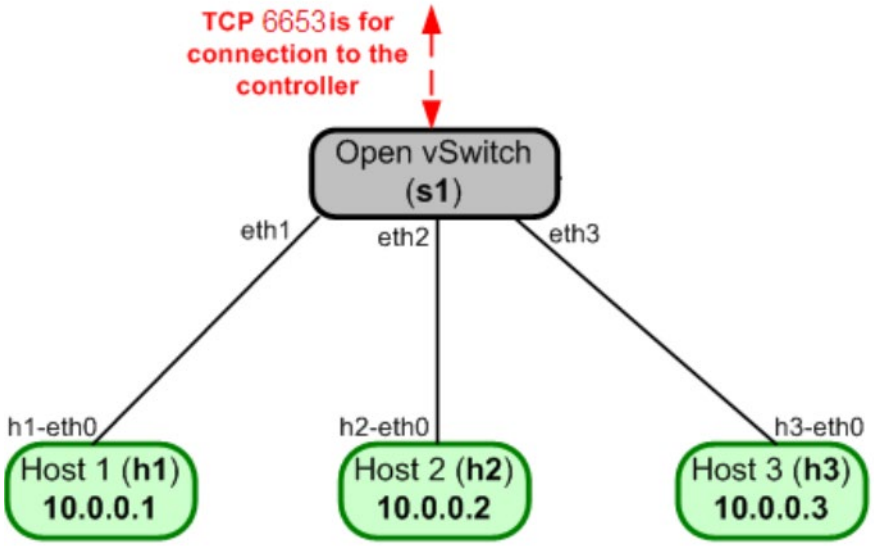
- دارای یک رابط Loopback Lo با آدرس IP = 127.0.0.1 پیکربندی شده است.
- دارای یک رابط s1-eth1 که آدرس IP ندارد.
- دارای یک رابط s1-eth2 است که آدرس IP ندارد.
- دارای یک رابط s1-eth3 که آدرس IP ندارد.
- شناسه فرآیند 50873 است.

❖ کنترلر c0 روی آدرس IP = 127.0.0.1 با پورت ۶۶۵۳ اجرا می شود. همچنین c0_PID = 50857 است.

- برای بررسی PIDهای h1، h2، h3، s1، c0، می توانیم از دستور ps -aux استفاده کنیم:

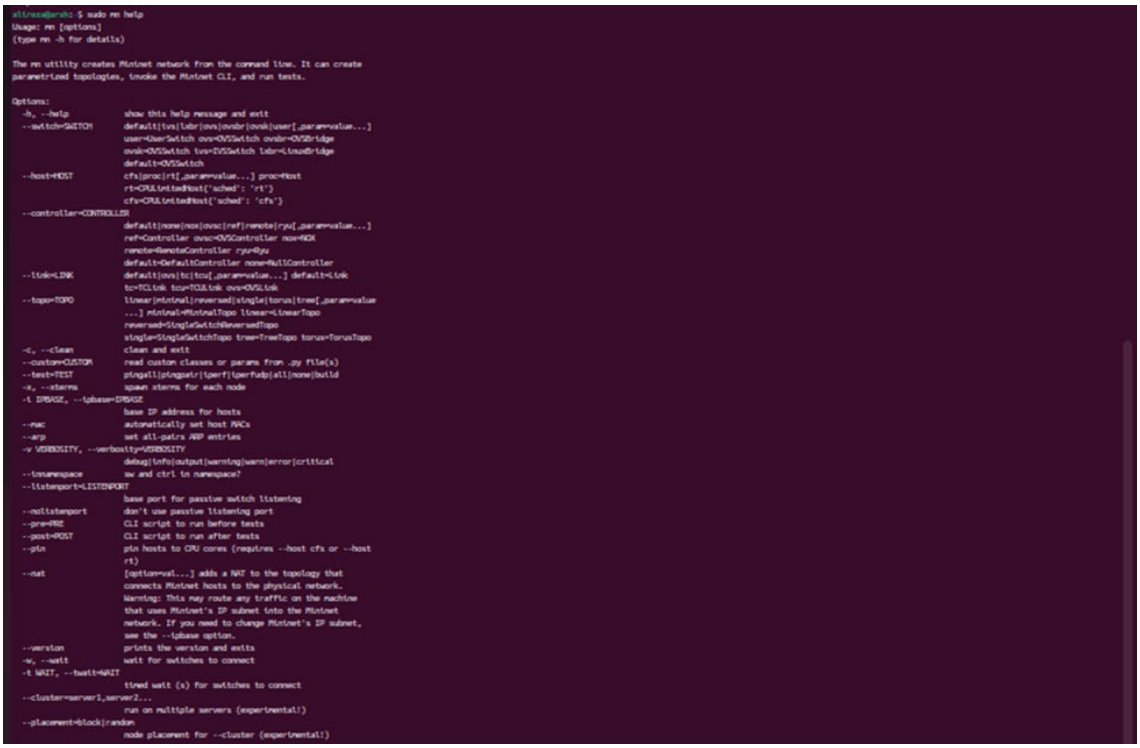
```
root       50857  0.0  0.0  21508  3932 pts/2    Ss+  11:18   0:00 bash --norc
root       50864  0.0  0.1  21508  4052 pts/3    Ss+  11:18   0:00 bash --norc
root       50866  0.0  0.0  21508  3940 pts/4    Ss+  11:18   0:00 bash --norc
root       50868  0.0  0.1  21508  4004 pts/5    Ss+  11:18   0:00 bash --norc
root       50873  0.0  0.1  21508  4052 pts/6    Ss+  11:18   0:00 bash --norc
```

- شکل کلی شبکه:



۳.

- استفاده از sudo mn -help:

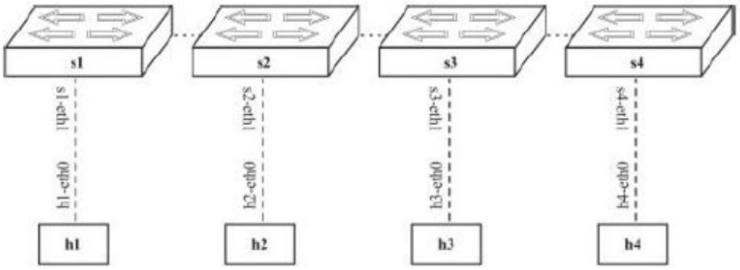


- انواع توپولوژی آماده:

```
  --topo=TOPO           linear|minimal|reversed|single|torus|tree[,param=value
                        ...] minimal=MinimalTopo linear=LinearTopo
                        reversed=SingleSwitchReversedTopo
                        single=SingleSwitchTopo tree=TreeTopo torus=TorusTopo
```

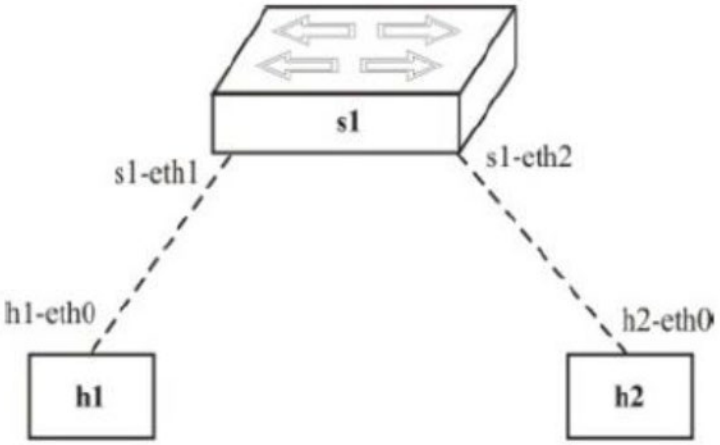
به ترتیب عبارتند از: Linear ، minimal ، reversed ، single ، torust ، tree

- Linear(خطی): توپولوژی خطی شامل k سوئیچ و k میزبان است. همچنین یک پیوند بین هر سوئیچ و هر میزبان و بین سوئیچ ها ایجاد می کند.



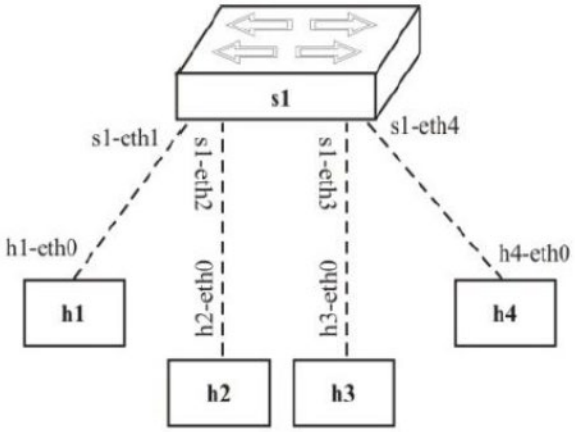

```
mininet> net
h1 h1-eth0:s1-eth1
h2 h2-eth0:s2-eth1
h3 h3-eth0:s3-eth1
h4 h4-eth0:s4-eth1
s1 lo:  s1-eth1:h1-eth0 s1-eth2:s2-eth2
s2 lo:  s2-eth1:h2-eth0 s2-eth2:s1-eth2 s2-eth3:s3-eth2
s3 lo:  s3-eth1:h3-eth0 s3-eth2:s2-eth3 s3-eth3:s4-eth2
s4 lo:  s4-eth1:h4-eth0 s4-eth2:s3-eth3
c0
```

- Minimal : یک توپولوژی بسیار ساده است که شامل یک سوئیچ Open Flow و ۲ میزبان است. همچنین پیوندهایی بین سوئیچ و دو میزبان ایجاد می کند.



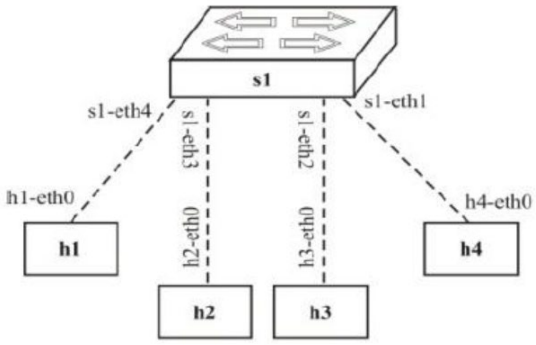

```
mininet> net
h1 h1-eth0:s1-eth1
h2 h2-eth0:s1-eth2
s1 lo:  s1-eth1:h1-eth0 s1-eth2:h2-eth0
c0
```

- Single: یک توپولوژی ساده با یک سوئیچ جریان باز و k میزبان است. همچنین یک پیوند بین سوئیچ و k میزبان ایجاد می کند.



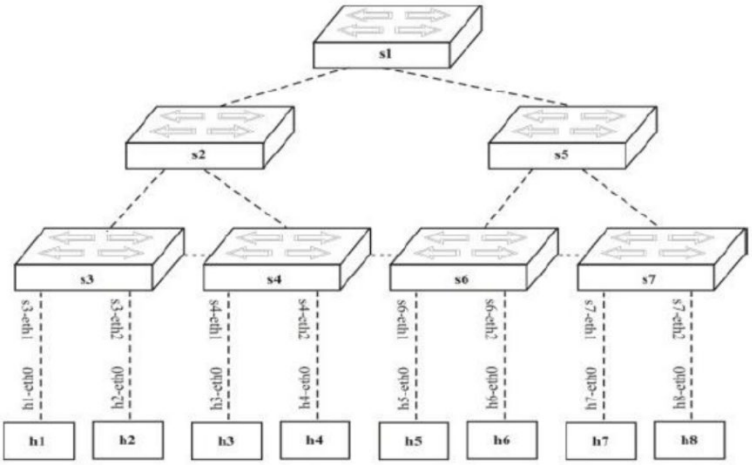

```
mininet> net
h1 h1-eth0:s1-eth1
h2 h2-eth0:s1-eth2
h3 h3-eth0:s1-eth3
h4 h4-eth0:s1-eth4
s1 lo:  s1-eth1:h1-eth0 s1-eth2:h2-eth0 s1-eth3:h3-eth0 s1-eth4:h4-eth0
c0
```

- Reversed: شبیه توپولوژی single است اما ترتیب اتصال معکوس است.




```
mininet> net
h1 h1-eth0:s1-eth4
h2 h2-eth0:s1-eth3
h3 h3-eth0:s1-eth2
h4 h4-eth0:s1-eth1
s1 lo:  s1-eth1:h4-eth0 s1-eth2:h3-eth0 s1-eth3:h2-eth0 s1-eth4:h1-eth0
c0
```

- درخت: توپولوژی درختی شامل k سطح است و ۲ میزبان به هر سوئیچ متصل است.




```
mininet> net
h1 h1-eth0:s3-eth1
h2 h2-eth0:s3-eth2
h3 h3-eth0:s4-eth1
h4 h4-eth0:s4-eth2
h5 h5-eth0:s6-eth1
h6 h6-eth0:s6-eth2
h7 h7-eth0:s7-eth1
h8 h8-eth0:s7-eth2
s1 lo:  s1-eth1:s2-eth3 s1-eth2:s5-eth3
s2 lo:  s2-eth1:s3-eth3 s2-eth2:s4-eth3 s2-eth3:s1-eth1
s3 lo:  s3-eth1:h1-eth0 s3-eth2:h2-eth0 s3-eth3:s2-eth1
s4 lo:  s4-eth1:h3-eth0 s4-eth2:h4-eth0 s4-eth3:s2-eth2
s5 lo:  s5-eth1:s6-eth3 s5-eth2:s7-eth3 s5-eth3:s1-eth2
s6 lo:  s6-eth1:h5-eth0 s6-eth2:h6-eth0 s6-eth3:s5-eth1
s7 lo:  s7-eth1:h7-eth0 s7-eth2:h8-eth0 s7-eth3:s5-eth2
c0
```

در توپولوژی درختی نیز می‌توانیم تعداد فرزندان در هر گره و عمق درخت را با استفاده از پارامترهای [depth] و [fanout] تعیین کنیم.

- پیاده سازی‌های هر یک از انواع توپولوژی‌های آماده در mininet در این لینک موجود می‌باشد.