



Trabalho de C115 Mininet - Trabalho Final

Gualter Machado Mesquita - 1601 - GEC



Trabalho Parte 1 - Topologia linear com 6 hosts

- Com uso de linha de comando padrão do Mininet, crie a topologia considerando o endereço MAC padronizado, larguras de banda bw de 5Mbps e controlador do Mininet (não precisa especificar).
- Inspeccione informações das interfaces, endereços MAC, IP e portas através de linhas de comando.
- Crie um desenho ilustrativo da topologia com todas as informações obtidas no item anterior
- Execute testes de ping entre os diferentes nós, mostre os pacotes chegando nos nós com uso do comando tcpdump.
- Especifique que o host 1 na porta 5555 vai ser um servidor TCP e o host 2 um cliente e execute testes de iperf, considere um relatório por segundo com teste de 15 segundos. Faça os testes para larguras de banda bw de 2, 10, 15 e 20 Mbps.

Topologia

[illegible]



Nós da topologia

```
mininet> nodes
available nodes are:
c0 h1 h2 h3 h4 h5 h6 s1 s2 s3 s4 s5 s6
```



Informações das interfaces

```
mininet> dump
<Host h1: h1-eth0:10.0.0.1 pid=15291>
<Host h2: h2-eth0:10.0.0.2 pid=15294>
<Host h3: h3-eth0:10.0.0.3 pid=15297>
<Host h4: h4-eth0:10.0.0.4 pid=15300>
<Host h5: h5-eth0:10.0.0.5 pid=15303>
<Host h6: h6-eth0:10.0.0.6 pid=15306>
<OVSSwitch s1: lo:127.0.0.1,s1-eth1:None,s1-eth2:None pid=15312>
<OVSSwitch s2: lo:127.0.0.1,s2-eth1:None,s2-eth2:None,s2-eth3:None pid=15315>
<OVSSwitch s3: lo:127.0.0.1,s3-eth1:None,s3-eth2:None,s3-eth3:None pid=15318>
<OVSSwitch s4: lo:127.0.0.1,s4-eth1:None,s4-eth2:None,s4-eth3:None pid=15321>
<OVSSwitch s5: lo:127.0.0.1,s5-eth1:None,s5-eth2:None,s5-eth3:None pid=15324>
<OVSSwitch s6: lo:127.0.0.1,s6-eth1:None,s6-eth2:None pid=15327>
<Controller c0: 127.0.0.1:6653 pid=15284>
mininet>
```



Conexão entre as interfaces

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



Interfaces, endereços de IP e MAC

```
mininet> h1 ifconfig -a
h1-eth0  Link encap:Ethernet  HWaddr 00:00:00:00:00:01
         inet addr:10.0.0.1  Bcast:10.255.255.255  Mask:255.0.0.0
         UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
         RX packets:0 errors:0 dropped:0 overruns:0 frame:0
         TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

lo       Link encap:Local Loopback
         inet addr:127.0.0.1  Mask:255.0.0.0
         UP LOOPBACK RUNNING  MTU:65536  Metric:1
         RX packets:0 errors:0 dropped:0 overruns:0 frame:0
         TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1
         RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

mininet>
```

Host h1 possui endereços IP e MAC terminados em 1



Interfaces, endereços de IP e MAC

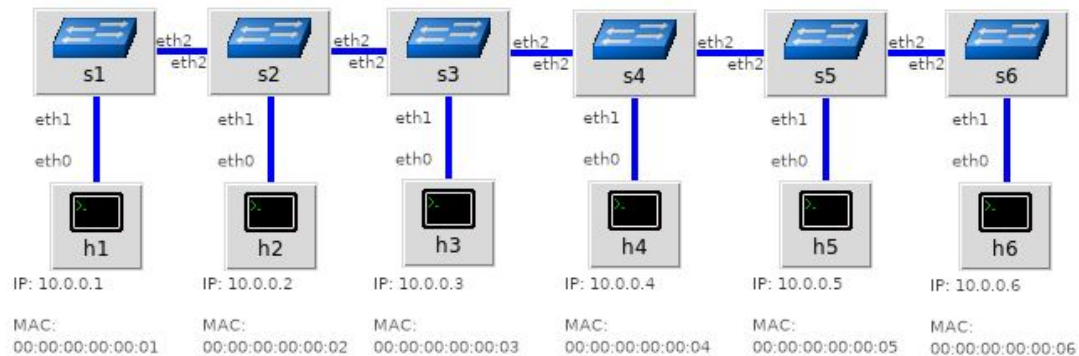
```
mininet> h6 ifconfig -a
h6-eth0  Link encap:Ethernet  HWaddr 00:00:00:00:00:06
         inet addr:10.0.0.6  Bcast:10.255.255.255  Mask:255.0.0.0
         UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
         RX packets:0 errors:0 dropped:0 overruns:0 frame:0
         TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

lo       Link encap:Local Loopback
         inet addr:127.0.0.1  Mask:255.0.0.0
         UP LOOPBACK RUNNING  MTU:65536  Metric:1
         RX packets:0 errors:0 dropped:0 overruns:0 frame:0
         TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1
         RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

mininet>
```

Host h6 possui endereços IP e MAC terminados em 6

Ilustração da topologia



Ping de h1 para h6

H6 mostra o fluxo da rede através do
TCPDUMP.

Node: h1

```
root@mininet-vn:~# ping 10.0.0.6
PING 10.0.0.6 (10.0.0.6) 56(84) bytes of data.
64 bytes from 10.0.0.6: icmp_seq=1 ttl=64 time=1.88 ms
64 bytes from 10.0.0.6: icmp_seq=2 ttl=64 time=0.139 ms
64 bytes from 10.0.0.6: icmp_seq=3 ttl=64 time=0.155 ms
^C
--- 10.0.0.6 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 0.139/0.727/1.888/0.821 ms
root@mininet-vn:~#
```

Node: h6

```
root@mininet-vn:~# tcpdump -XX -n -i h6-eth0
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on h6-eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
17:44:23.218490 IP 10.0.0.1 > 10.0.0.6: ICMP echo request, id 17198, seq 1, leng
th 64
0x0000: 0000 0000 0006 0000 0000 0001 0800 4500 .....E.
0x0010: 0054 2280 4000 4001 0423 0a00 0001 0a00 .T..@..#.....
0x0020: 0006 0800 7377 432e 0001 67cd 3165 0000 ....suC...g.i.e..
0x0030: 0000 e653 0300 0000 0000 1011 1213 1415 ..S.....
0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....!"#%&
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637 67
17:44:23.218493 IP 10.0.0.6 > 10.0.0.1: ICMP echo reply, id 17198, seq 1, length
64
0x0000: 0000 0000 0001 0000 0000 0006 0800 4500 .....E.
0x0010: 0054 0809 0000 4001 5e9a 0a00 0006 0a00 .T...@..#.....
0x0020: 0001 0000 7b77 432e 0001 67cd 3165 0000 ....(uC...g.i.e..
0x0030: 0000 e653 0300 0000 0000 1011 1213 1415 ..S.....
0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....!"#%&
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637
17:44:24.220811 IP 10.0.0.1 > 10.0.0.6: ICMP echo request, id 17198, seq 2, leng
th 64
0x0000: 0000 0000 0006 0000 0000 0001 0800 4500 .....E.
0x0010: 0054 2291 4000 4001 0412 0a00 0001 0a00 .T..@&.....
0x0020: 0006 0800 f46b 432e 0002 68cd 3165 0000 ....kC...h.i.e..
0x0030: 0000 645e 0300 0000 0000 1011 1213 1415 ..d.....
0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....!"#%&
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637
17:44:24.220895 IP 10.0.0.6 > 10.0.0.1: ICMP echo reply, id 17198, seq 2, length
64
0x0000: 0000 0000 0001 0000 0000 0006 0800 4500 .....E.
0x0010: 0054 0818 0000 4001 5e8b 0a00 0006 0a00 .T...@..#.....
0x0020: 0001 0000 fe6b 432e 0002 68cd 3165 0000 ....kC...h.i.e..
0x0030: 0000 645e 0300 0000 0000 1011 1213 1415 ..d.....
0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....!"#%&
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637
17:44:25.221601 IP 10.0.0.1 > 10.0.0.6: ICMP echo request, id 17198, seq 3, leng
th 64
0x0000: 0000 0000 0006 0000 0000 0001 0800 4500 .....E.
0x0010: 0054 22e1 4000 4001 03c2 0a00 0001 0a00 .T..@&.....
0x0020: 0006 0800 dc67 432e 0003 68cd 3165 0000 ....gC...i.i.e..
0x0030: 0000 7b61 0300 0000 0000 1011 1213 1415 ..(a.....
0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....!"#%&
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637
17:44:25.221703 IP 10.0.0.6 > 10.0.0.1: ICMP echo reply, id 17198, seq 3, leng
th 64
0x0000: 0000 0000 0001 0000 0000 0006 0800 4500 .....E.
0x0010: 0054 0843 0000 4001 5e6b 0a00 0006 0a00 .T..@&.....
0x0020: 0001 0000 e467 432e 0003 68cd 3165 0000 ....gC...i.i.e..
0x0030: 0000 7b61 0300 0000 0000 1011 1213 1415 ..(a.....
0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....!"#%&
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637
```

Ping de h2 para h3

H3 mostra o fluxo da rede através do
TCPDUMP.

h2

```
root@mininet-vn:~# ping 10.0.0.3
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=10.8 ms
64 bytes from 10.0.0.3: icmp_seq=2 ttl=64 time=0.375 ms
64 bytes from 10.0.0.3: icmp_seq=3 ttl=64 time=0.196 ms
^C
--- 10.0.0.3 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 0.196/3.818/10.885/4.997 ms
root@mininet-vn:~#
```

h3

```
root@mininet-vn:~# tcpdump -XX -n -i h3-eth0
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on h3-eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
17:48:34.219246 ARP, Request who-has 10.0.0.3 tell 10.0.0.2, length 28
0x0000: ffff ffff ffff 0000 0000 0002 0806 0001 .....
0x0010: 0800 0604 0001 0000 0000 0002 0a00 0002 .....
0x0020: 0000 0000 0000 0a00 0003 .....
17:48:34.219259 ARP, Reply 10.0.0.3 is-at 00:00:00:00:00:03, length 28
0x0000: 0000 0000 0002 0000 0000 0003 0806 0001 .....
0x0010: 0800 0604 0002 0000 0000 0003 0a00 0003 .....
0x0020: 0000 0000 0002 0a00 0002 .....
17:48:34.224676 IP 10.0.0.2 > 10.0.0.3: ICMP echo request, id 17233, seq 1, length 64
0x0000: 0000 0000 0003 0000 0000 0002 0800 4500 .....E.
0x0010: 0054 123d 4000 4001 1408 0a00 0002 0a00 ....T..h..
0x0020: 0003 0800 ea56 4351 0001 62ce 3165 0000 ....VC00..b.i.e..
0x0030: 0000 7450 0300 0000 0000 1011 1213 1415 ...+V.....
0x0040: 1817 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....I*%$Z
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637 .....67
17:48:34.224968 IP 10.0.0.3 > 10.0.0.2: ICMP echo reply, id 17233, seq 1, length 64
0x0000: 0000 0000 0002 0000 0000 0003 0800 4500 .....E.
0x0010: 0054 fd68 0000 4001 693c 0a00 0002 0a00 ....T..h..i.k.....
0x0020: 0002 0000 f256 4351 0001 62ce 3165 0000 ....VC00..b.i.e..
0x0030: 0000 7450 0300 0000 0000 1011 1213 1415 ...+V.....
0x0040: 1817 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....I*%$Z
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637 .....67
17:48:35.219741 IP 10.0.0.2 > 10.0.0.3: ICMP echo request, id 17233, seq 2, length 64
0x0000: 0000 0000 0003 0000 0000 0002 0800 4500 .....E.
0x0010: 0054 12bf 4000 4001 136e 0a00 0002 0a00 ....T..h..c.i.e..
0x0020: 0003 0800 324d 4351 0002 63ce 3165 0000 ....2M00..c.i.e..
0x0030: 0000 2b59 0300 0000 0000 1011 1213 1415 ...+V.....
0x0040: 1817 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....I*%$Z
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637 .....67
17:48:35.219756 IP 10.0.0.3 > 10.0.0.2: ICMP echo reply, id 17233, seq 2, length 64
0x0000: 0000 0000 0002 0000 0000 0003 0800 4500 .....E.
0x0010: 0054 fda5 0000 4001 68ff 0a00 0002 0a00 ....T..h..h.....
0x0020: 0002 0000 344d 4351 0002 63ce 3165 0000 ....2M00..c.i.e..
0x0030: 0000 2b59 0300 0000 0000 1011 1213 1415 ...+V.....
0x0040: 1817 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....I*%$Z
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637 .....67
17:48:36.218461 IP 10.0.0.2 > 10.0.0.3: ICMP echo request, id 17233, seq 3, length 64
0x0000: 0000 0000 0003 0000 0000 0002 0800 4500 .....E.
0x0010: 0054 132b 4000 4001 137a 0a00 0002 0a00 ....T..h..l.z.....
0x0020: 0003 0800 1850 4351 0003 64ce 3165 0000 ....PC00..d.i.e..
0x0030: 0000 4455 0300 0000 0000 1011 1213 1415 ...BU.....
0x0040: 1817 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....I*%$Z
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637 .....67
17:48:36.218577 IP 10.0.0.3 > 10.0.0.2: ICMP echo reply, id 17233, seq 3, length 64
0x0000: 0000 0000 0002 0000 0000 0003 0800 4500 .....E.
0x0010: 0054 fe85 0000 4001 681f 0a00 0002 0a00 ....T..h..h.....
0x0020: 0002 0000 2050 4351 0003 64ce 3165 0000 ....PC00..d.i.e..
0x0030: 0000 4455 0300 0000 0000 1011 1213 1415 ...BU.....
0x0040: 1817 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....I*%$Z
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637 .....67
```

Ping de h4 para h5

H5 mostra o fluxo da rede através do TCPDUMP.

```
*Node: h4*
root@mininet-v4:~# ping 10.0.0.5
PING 10.0.0.5 (10.0.0.5) 56(84) bytes of data:
64 bytes from 10.0.0.5: icmp_seq=1 ttl=64 time=13.3 ms
64 bytes from 10.0.0.5: icmp_seq=2 ttl=64 time=1.31 ms
64 bytes from 10.0.0.5: icmp_seq=3 ttl=64 time=0.043 ms
^C
--- 10.0.0.5 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2038ms
rtt min/avg/max/ndev = 0.043/5.121/13.998/6.298 ms
root@mininet-v4:~#
```

```
*Node: h5*
root@mininet-v4:~# tcpdump -XX -n -i h5-eth0
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on h5-eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
17:50:46.382746 HFP, Request who-has 10.0.0.5 tell 10.0.0.4, length 28
0x0000: ffff ffff ffff 0000 0000 0004 0806 0001 .....
0x0010: 0800 0604 0001 0000 0000 0004 0a00 0004 .....
0x0020: 0000 0000 0000 0a00 0005 .....
17:50:46.382832 HFP, Reply 10.0.0.5 is-at 00:00:00:00:00:05, length 28
0x0000: 0000 0000 0004 0000 0000 0005 0806 0001 .....
0x0010: 0800 0604 0002 0000 0000 0005 0a00 0005 .....
0x0020: 0000 0000 0004 0a00 0004 .....
17:50:46.391303 IP 10.0.0.4 > 10.0.0.5: ICMP echo request, id 17280, seq 1, length 64
0x0000: 0000 0000 0005 0000 0000 0004 0800 4500 .....E.
0x0010: 0054 f055 4000 4001 364b 0a00 0004 0a00 ..T.U@.@.6K.....
0x0020: 0005 0800 1da9 4380 0001 e6ce 3165 0000 .....C.....ie..
0x0030: 0000 bace 0500 0000 0000 1011 1213 1415 .....
0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....!"%$%
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637 .....67
17:50:46.391418 IP 10.0.0.5 > 10.0.0.4: ICMP echo reply, id 17280, seq 1, length 64
0x0000: 0000 0000 0004 0000 0000 0005 0800 4500 .....E.
0x0010: 0054 bfb1 0000 4001 a6ef 0a00 0005 0a00 ..T...@.....
0x0020: 0004 0000 25a9 4380 0001 e6ce 3165 0000 .....%C.....ie..
0x0030: 0000 bace 0500 0000 0000 1011 1213 1415 .....
0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....!"%$%
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637 .....67
17:50:47.418593 IP 10.0.0.4 > 10.0.0.5: ICMP echo request, id 17280, seq 2, length 64
0x0000: 0000 0000 0005 0000 0000 0004 0800 4500 .....E.
0x0010: 0054 f12d 4000 4001 3573 0a00 0004 0a00 ..T.-@.@.5s.....
0x0020: 0005 0800 ff17 4380 0002 e7ce 3165 0000 .....C.....ie..
0x0030: 0000 d75e 0600 0000 0000 1011 1213 1415 .....
0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....!"%$%
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637 .....67
17:50:47.418674 IP 10.0.0.5 > 10.0.0.4: ICMP echo reply, id 17280, seq 2, length 64
0x0000: 0000 0000 0004 0000 0000 0005 0800 4500 .....E.
0x0010: 0054 c012 0000 4001 a68e 0a00 0005 0a00 ..T...@.....
0x0020: 0004 0000 0718 4380 0002 e7ce 3165 0000 .....C.....ie..
0x0030: 0000 d75e 0600 0000 0000 1011 1213 1415 .....
0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....!"%$%
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637 .....67
17:50:48.418888 IP 10.0.0.4 > 10.0.0.5: ICMP echo request, id 17280, seq 3, length 64
0x0000: 0000 0000 0005 0000 0000 0004 0800 4500 .....E.
0x0010: 0054 f168 4000 4001 3538 0a00 0004 0a00 ..T.h@.@.58.....
0x0020: 0005 0800 a911 4380 0003 a8ce 3165 0000 .....C.....ie..
0x0030: 0000 2c64 0600 0000 0000 1011 1213 1415 .....d.....
0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....!"%$%
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637 .....67
17:50:48.418903 IP 10.0.0.5 > 10.0.0.4: ICMP echo reply, id 17280, seq 3, length 64
0x0000: 0000 0000 0004 0000 0000 0005 0800 4500 .....E.
0x0010: 0054 c040 0000 4001 a690 0a00 0005 0a00 ..T.@.@.....
0x0020: 0004 0000 b111 4380 0003 a8ce 3165 0000 .....C.....ie..
0x0030: 0000 2c64 0600 0000 0000 1011 1213 1415 .....d.....
0x0040: 1617 1819 1a1b 1c1d 1e1f 2021 2223 2425 .....!"%$%
0x0050: 2627 2829 2a2b 2c2d 2e2f 3031 3233 3435 &'()*+,-./012345
0x0060: 3637 .....67
```

Simulação Servidor/Cliente TCP: bw = 2 Mbps

"Node: h1"

```
root@mininet-virtual-machine:~# iperf -s -p 5555 -i 1
```

```
-----  
Server listening on TCP port 5555  
TCP window size: 85.3 KByte (default)  
-----
```

```
[ 32] local 10.0.0.1 port 5555 connected with 10.0.0.2 port 35388  
[ ID] Interval      Transfer    Bandwidth  
[ 32] 0.0- 1.0 sec   235 KBytes  1.92 Mbits/sec  
[ 32] 1.0- 2.0 sec   229 KBytes  1.88 Mbits/sec  
[ 32] 2.0- 3.0 sec   228 KBytes  1.87 Mbits/sec  
[ 32] 3.0- 4.0 sec   232 KBytes  1.90 Mbits/sec  
[ 32] 4.0- 5.0 sec   232 KBytes  1.90 Mbits/sec  
[ 32] 5.0- 6.0 sec   233 KBytes  1.91 Mbits/sec  
[ 32] 6.0- 7.0 sec   235 KBytes  1.92 Mbits/sec  
[ 32] 7.0- 8.0 sec   232 KBytes  1.90 Mbits/sec  
[ 32] 8.0- 9.0 sec   233 KBytes  1.91 Mbits/sec  
[ 32] 9.0-10.0 sec   232 KBytes  1.90 Mbits/sec  
[ 32] 10.0-11.0 sec   232 KBytes  1.90 Mbits/sec  
[ 32] 11.0-12.0 sec   228 KBytes  1.87 Mbits/sec  
[ 32] 12.0-13.0 sec   229 KBytes  1.88 Mbits/sec  
[ 32] 13.0-14.0 sec   230 KBytes  1.89 Mbits/sec  
[ 32] 14.0-15.0 sec   229 KBytes  1.88 Mbits/sec  
[ 32] 15.0-16.0 sec   219 KBytes  1.80 Mbits/sec  
[ 32] 16.0-17.0 sec   143 KBytes  1.17 Mbits/sec  
[ 32] 0.0-17.0 sec   3.75 MBytes  1.85 Mbits/sec
```

"Node: h2"

```
root@mininet-virtual-machine:~# iperf -c 10.0.0.1 -p 5555 -i 1 -t15
```

```
-----  
Client connecting to 10.0.0.1, TCP port 5555  
TCP window size: 85.3 KByte (default)  
-----
```

```
[ 31] local 10.0.0.2 port 35388 connected with 10.0.0.1 port 5555  
[ ID] Interval      Transfer    Bandwidth  
[ 31] 0.0- 1.0 sec   512 KBytes  4.19 Mbits/sec  
[ 31] 1.0- 2.0 sec   256 KBytes  2.10 Mbits/sec  
[ 31] 2.0- 3.0 sec   128 KBytes  1.05 Mbits/sec  
[ 31] 3.0- 4.0 sec   256 KBytes  2.10 Mbits/sec  
[ 31] 4.0- 5.0 sec   128 KBytes  1.05 Mbits/sec  
[ 31] 5.0- 6.0 sec   256 KBytes  2.10 Mbits/sec  
[ 31] 6.0- 7.0 sec   256 KBytes  2.10 Mbits/sec  
[ 31] 7.0- 8.0 sec   256 KBytes  2.10 Mbits/sec  
[ 31] 8.0- 9.0 sec   256 KBytes  2.10 Mbits/sec  
[ 31] 9.0-10.0 sec   256 KBytes  2.10 Mbits/sec  
[ 31] 10.0-11.0 sec   128 KBytes  1.05 Mbits/sec  
[ 31] 11.0-12.0 sec   256 KBytes  2.10 Mbits/sec  
[ 31] 12.0-13.0 sec   256 KBytes  2.10 Mbits/sec  
[ 31] 13.0-14.0 sec   256 KBytes  2.10 Mbits/sec  
[ 31] 14.0-15.0 sec   256 KBytes  2.10 Mbits/sec  
[ 31] 0.0-16.1 sec   3.75 MBytes  1.96 Mbits/sec
```

```
root@mininet-virtual-machine:~#
```

Simulação Servidor/Cliente TCP: bw = 10 Mbps

"Node: h1"

root@mininet-vm:~# iperf -s -p 5555 -i 1

Server listening on TCP port 5555
TCP window size: 85.3 KByte (default)

[32]	local 10.0.0.1	port 5555	connected with 10.0.0.2	port 35536
[ID]	Interval	Transfer	Bandwidth	
[32]	0.0- 1.0 sec	826 KBytes	6.77 Mbits/sec	
[32]	1.0- 2.0 sec	769 KBytes	6.30 Mbits/sec	
[32]	2.0- 3.0 sec	689 KBytes	5.64 Mbits/sec	
[32]	3.0- 4.0 sec	905 KBytes	7.41 Mbits/sec	
[32]	4.0- 5.0 sec	935 KBytes	7.66 Mbits/sec	
[32]	5.0- 6.0 sec	969 KBytes	7.94 Mbits/sec	
[32]	6.0- 7.0 sec	987 KBytes	8.09 Mbits/sec	
[32]	7.0- 8.0 sec	950 KBytes	7.78 Mbits/sec	
[32]	8.0- 9.0 sec	979 KBytes	8.02 Mbits/sec	
[32]	9.0-10.0 sec	936 KBytes	7.67 Mbits/sec	
[32]	10.0-11.0 sec	660 KBytes	5.41 Mbits/sec	
[32]	11.0-12.0 sec	184 KBytes	1.51 Mbits/sec	
[32]	12.0-13.0 sec	180 KBytes	1.47 Mbits/sec	
[32]	13.0-14.0 sec	146 KBytes	1.19 Mbits/sec	
[32]	14.0-15.0 sec	291 KBytes	2.39 Mbits/sec	
[32]	15.0-16.0 sec	136 KBytes	1.11 Mbits/sec	
[32]	16.0-17.0 sec	467 KBytes	3.82 Mbits/sec	
[32]	0.0-17.2 sec	10.8 MBytes	5.23 Mbits/sec	

"Node: h2"

root@mininet-vm:~# iperf -c 10.0.0.1 -p 5555 -i 1 -t15

Client connecting to 10.0.0.1, TCP port 5555
TCP window size: 85.3 KByte (default)

[31]	local 10.0.0.2	port 35536	connected with 10.0.0.1	port 5555
[ID]	Interval	Transfer	Bandwidth	
[31]	0.0- 1.0 sec	1.12 MBytes	9.44 Mbits/sec	
[31]	1.0- 2.0 sec	768 KBytes	6.29 Mbits/sec	
[31]	2.0- 3.0 sec	768 KBytes	6.29 Mbits/sec	
[31]	3.0- 4.0 sec	1.00 MBytes	8.39 Mbits/sec	
[31]	4.0- 5.0 sec	896 KBytes	7.34 Mbits/sec	
[31]	5.0- 6.0 sec	1.00 MBytes	8.39 Mbits/sec	
[31]	6.0- 7.0 sec	896 KBytes	7.34 Mbits/sec	
[31]	7.0- 8.0 sec	1.00 MBytes	8.39 Mbits/sec	
[31]	8.0- 9.0 sec	896 KBytes	7.34 Mbits/sec	
[31]	9.0-10.0 sec	1.00 MBytes	8.39 Mbits/sec	
[31]	10.0-11.0 sec	512 KBytes	4.19 Mbits/sec	
[31]	11.0-12.0 sec	128 KBytes	1.05 Mbits/sec	
[31]	12.0-13.0 sec	256 KBytes	2.10 Mbits/sec	
[31]	13.0-14.0 sec	128 KBytes	1.05 Mbits/sec	
[31]	14.0-15.0 sec	384 KBytes	3.15 Mbits/sec	
[31]	15.0-16.0 sec	0.00 Bytes	0.00 bits/sec	
[31]	0.0-16.3 sec	10.8 MBytes	5.55 Mbits/sec	

root@mininet-vm:~#

Simulação Servidor/Cliente TCP: bw = 15 Mbps



"Node: h1"



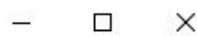
```
root@mininet-vm:~# iperf -s -p 5555 -i 1
```

```
-----  
Server listening on TCP port 5555  
TCP window size: 85.3 KByte (default)  
-----
```

```
[ 32] local 10.0.0.1 port 5555 connected with 10.0.0.2 port 35496  
[ ID] Interval      Transfer    Bandwidth  
[ 32] 0.0- 1.0 sec  1.68 MBytes 14.1 Mbits/sec  
[ 32] 1.0- 2.0 sec  1.66 MBytes 13.9 Mbits/sec  
[ 32] 2.0- 3.0 sec  1.67 MBytes 14.0 Mbits/sec  
[ 32] 3.0- 4.0 sec  1.66 MBytes 14.0 Mbits/sec  
[ 32] 4.0- 5.0 sec  1.67 MBytes 14.0 Mbits/sec  
[ 32] 5.0- 6.0 sec  1.69 MBytes 14.2 Mbits/sec  
[ 32] 6.0- 7.0 sec  1.67 MBytes 14.0 Mbits/sec  
[ 32] 7.0- 8.0 sec  1.66 MBytes 13.9 Mbits/sec  
[ 32] 8.0- 9.0 sec  1.56 MBytes 13.1 Mbits/sec  
[ 32] 9.0-10.0 sec   699 KBytes  5.72 Mbits/sec  
[ 32] 10.0-11.0 sec   882 KBytes  7.23 Mbits/sec  
[ 32] 11.0-12.0 sec   1.00 MBytes  8.42 Mbits/sec  
[ 32] 12.0-13.0 sec   908 KBytes  7.44 Mbits/sec  
[ 32] 13.0-14.0 sec   1.58 MBytes 13.2 Mbits/sec  
[ 32] 14.0-15.0 sec   1.67 MBytes 14.0 Mbits/sec  
[ 32] 0.0-15.8 sec  21.9 MBytes 11.6 Mbits/sec
```



"Node: h2"



```
root@mininet-vm:~# iperf -c 10.0.0.1 -p 5555 -i 1 -t15
```

```
-----  
Client connecting to 10.0.0.1, TCP port 5555  
TCP window size: 85.3 KByte (default)  
-----
```

```
[ 31] local 10.0.0.2 port 35496 connected with 10.0.0.1 port 5555  
[ ID] Interval      Transfer    Bandwidth  
[ 31] 0.0- 1.0 sec  1.88 MBytes 15.7 Mbits/sec  
[ 31] 1.0- 2.0 sec  1.62 MBytes 13.6 Mbits/sec  
[ 31] 2.0- 3.0 sec  1.62 MBytes 13.6 Mbits/sec  
[ 31] 3.0- 4.0 sec  1.75 MBytes 14.7 Mbits/sec  
[ 31] 4.0- 5.0 sec  1.62 MBytes 13.6 Mbits/sec  
[ 31] 5.0- 6.0 sec  1.75 MBytes 14.7 Mbits/sec  
[ 31] 6.0- 7.0 sec  1.62 MBytes 13.6 Mbits/sec  
[ 31] 7.0- 8.0 sec  1.75 MBytes 14.7 Mbits/sec  
[ 31] 8.0- 9.0 sec  1.50 MBytes 12.6 Mbits/sec  
[ 31] 9.0-10.0 sec   640 KBytes  5.24 Mbits/sec  
[ 31] 10.0-11.0 sec   896 KBytes  7.34 Mbits/sec  
[ 31] 11.0-12.0 sec   1.00 MBytes  8.39 Mbits/sec  
[ 31] 12.0-13.0 sec   896 KBytes  7.34 Mbits/sec  
[ 31] 13.0-14.0 sec   1.62 MBytes 13.6 Mbits/sec  
[ 31] 14.0-15.0 sec   1.62 MBytes 13.6 Mbits/sec  
[ 31] 0.0-15.1 sec  21.9 MBytes 12.2 Mbits/sec
```

```
root@mininet-vm:~#
```

Simulação Servidor/Cliente TCP: bw = 20 Mbps

```
"Node: h1"
root@mininet-virtual-machine:~# iperf -s -p 5555 -i 1

-----
Server listening on TCP port 5555
TCP window size: 85.3 KByte (default)
-----

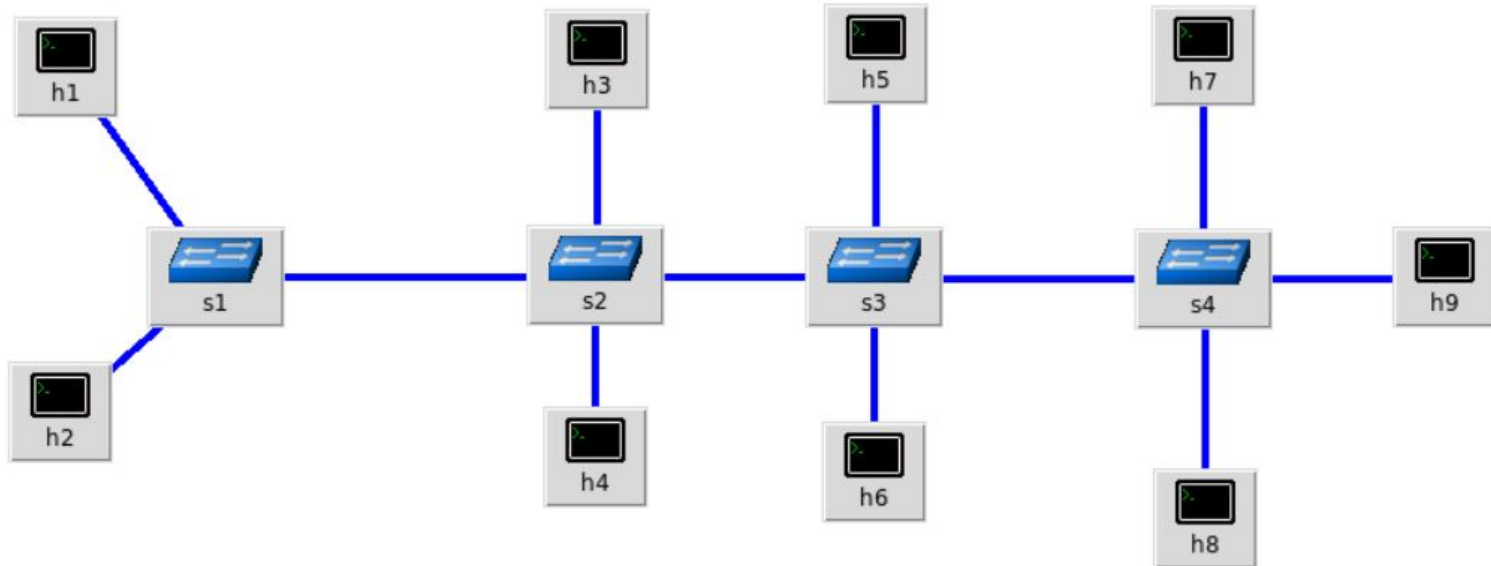
[ 32] local 10.0.0.1 port 5555 connected with 10.0.0.2 port 35560
[ ID] Interval      Transfer    Bandwidth
[ 32] 0.0- 1.0 sec    554 KBytes  4.54 Mbits/sec
[ 32] 1.0- 2.0 sec    810 KBytes  6.64 Mbits/sec
[ 32] 2.0- 3.0 sec    1.09 MBytes 9.13 Mbits/sec
[ 32] 3.0- 4.0 sec    1.22 MBytes 10.2 Mbits/sec
[ 32] 4.0- 5.0 sec    2.05 MBytes 17.2 Mbits/sec
[ 32] 5.0- 6.0 sec    2.18 MBytes 18.3 Mbits/sec
[ 32] 6.0- 7.0 sec    2.18 MBytes 18.3 Mbits/sec
[ 32] 7.0- 8.0 sec    2.18 MBytes 18.3 Mbits/sec
[ 32] 8.0- 9.0 sec    2.17 MBytes 18.2 Mbits/sec
[ 32] 9.0-10.0 sec    2.16 MBytes 18.1 Mbits/sec
[ 32] 10.0-11.0 sec   2.17 MBytes 18.2 Mbits/sec
[ 32] 11.0-12.0 sec   540 KBytes  4.43 Mbits/sec
[ 32] 12.0-13.0 sec   409 KBytes  3.35 Mbits/sec
[ 32] 13.0-14.0 sec   424 KBytes  3.48 Mbits/sec
[ 32] 14.0-15.0 sec   1.29 MBytes 10.8 Mbits/sec
[ 32] 0.0-15.2 sec   21.8 MBytes 12.0 Mbits/sec
```

```
"Node: h2"
root@mininet-virtual-machine:~# iperf -c 10.0.0.1 -p 5555 -i 1 -t15

-----
Client connecting to 10.0.0.1, TCP port 5555
TCP window size: 85.3 KByte (default)
-----

[ 31] local 10.0.0.2 port 35560 connected with 10.0.0.1 port 5555
[ ID] Interval      Transfer    Bandwidth
[ 31] 0.0- 1.0 sec    768 KBytes  6.29 Mbits/sec
[ 31] 1.0- 2.0 sec    768 KBytes  6.29 Mbits/sec
[ 31] 2.0- 3.0 sec    1.12 MBytes 9.44 Mbits/sec
[ 31] 3.0- 4.0 sec    1.25 MBytes 10.5 Mbits/sec
[ 31] 4.0- 5.0 sec    2.12 MBytes 17.8 Mbits/sec
[ 31] 5.0- 6.0 sec    2.12 MBytes 17.8 Mbits/sec
[ 31] 6.0- 7.0 sec    2.25 MBytes 18.9 Mbits/sec
[ 31] 7.0- 8.0 sec    2.00 MBytes 16.8 Mbits/sec
[ 31] 8.0- 9.0 sec    2.25 MBytes 18.9 Mbits/sec
[ 31] 9.0-10.0 sec    2.12 MBytes 17.8 Mbits/sec
[ 31] 10.0-11.0 sec   2.25 MBytes 18.9 Mbits/sec
[ 31] 11.0-12.0 sec   512 KBytes  4.19 Mbits/sec
[ 31] 12.0-13.0 sec   384 KBytes  3.15 Mbits/sec
[ 31] 13.0-14.0 sec   384 KBytes  3.15 Mbits/sec
[ 31] 14.0-15.0 sec   1.38 MBytes 11.5 Mbits/sec
[ 31] 0.0-15.1 sec   21.8 MBytes 12.1 Mbits/sec
root@mininet-virtual-machine:~#
```


Trabalho Parte 2 - Topologia customizada





Trabalho Parte 2 - Topologia customizada

- Com uso de linha de comando padrão do Mininet, crie a topologia customizada considerando o endereço MAC padronizado e controlador manual.
- Inspeccione informações das interfaces, endereços MAC, IP e portas através de linhas de comando.
- Crie um desenho ilustrativo da topologia com todas as informações obtidas no item anterior.
- Faça testes de ping considerando os switches normais.
- Apague as regras anteriores e crie regras baseadas em endereços MAC para alguns nós. (Deve-se comunicar hosts dos diferentes switches).
- Faça testes de ping para demonstrar que as regras foram bem implementadas.



Criação da Topologia em Python

```
from mininet.topo import Topo

class MyTopo(Topo):
    """
    MyTopo cria uma topologia de rede customizada, de acordo com a segunda parte do trabalho final de Mininet.
    """

    def __init__(self):
        # Inicializando a topologia
        Topo.__init__(self)

        # Adicionando os Hosts da topologia
        h1 = self.addHost('h1')
        h2 = self.addHost('h2')
        h3 = self.addHost('h3')
        h4 = self.addHost('h4')
        h5 = self.addHost('h5')
        h6 = self.addHost('h6')
        h7 = self.addHost('h7')
        h8 = self.addHost('h8')
        h9 = self.addHost('h9')

        # Adicionando os Switches da topologia
        s1 = self.addSwitch('s1')
        s2 = self.addSwitch('s2')
        s3 = self.addSwitch('s3')
        s4 = self.addSwitch('s4')
```



Criação da Topologia em Python

```
# Adicionando os links entre as máquinas
# H1, H2 -> S1
self.addLink(h1, s1)
self.addLink(h2, s1)

# H3, H4 -> S2
self.addLink(h3, s2)
self.addLink(h4, s2)

# H5, H6 -> S3
self.addLink(h5, s3)
self.addLink(h6, s3)

# H7, H8, H9 -> S4
self.addLink(h7, s4)
self.addLink(h8, s4)
self.addLink(h9, s4)

# Conexão entre os switches
self.addLink(s1, s2)
self.addLink(s2, s3)
self.addLink(s3, s4)

topos = {'mytopo': (lambda: MyTopo())}
```

Criação da Topologia no Mininet

```
mininet@mininet-vm:~$ sudo mn --custom Downloads/topo.py --topo mytopo --controller=none --mac
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2 h3 h4 h5 h6 h7 h8 h9
*** Adding switches:
s1 s2 s3 s4
*** Adding links:
(h1, s1) (h2, s1) (h3, s2) (h4, s2) (h5, s3) (h6, s3) (h7, s4) (h8, s4) (h9, s4) (s1, s2) (s2,
s3) (s3, s4)
*** Configuring hosts
h1 h2 h3 h4 h5 h6 h7 h8 h9
*** Starting controller

*** Starting 4 switches
s1 s2 s3 s4 ...
*** Starting CLI:
mininet>
```



Nós da topologia

```
mininet> nodes
available nodes are:
h1 h2 h3 h4 h5 h6 h7 h8 h9 s1 s2 s3 s4
```



Informações das interfaces

```
mininet> dump
<Host h1: h1-eth0:10.0.0.1 pid=23235>
<Host h2: h2-eth0:10.0.0.2 pid=23238>
<Host h3: h3-eth0:10.0.0.3 pid=23241>
<Host h4: h4-eth0:10.0.0.4 pid=23244>
<Host h5: h5-eth0:10.0.0.5 pid=23247>
<Host h6: h6-eth0:10.0.0.6 pid=23250>
<Host h7: h7-eth0:10.0.0.7 pid=23253>
<Host h8: h8-eth0:10.0.0.8 pid=23256>
<Host h9: h9-eth0:10.0.0.9 pid=23259>
<OVSSwitch s1: lo:127.0.0.1,s1-eth1:None,s1-eth2:None,s1-eth3:None pid=23265>
<OVSSwitch s2: lo:127.0.0.1,s2-eth1:None,s2-eth2:None,s2-eth3:None,s2-eth4:None pid=23268>
<OVSSwitch s3: lo:127.0.0.1,s3-eth1:None,s3-eth2:None,s3-eth3:None,s3-eth4:None pid=23271>
<OVSSwitch s4: lo:127.0.0.1,s4-eth1:None,s4-eth2:None,s4-eth3:None,s4-eth4:None pid=23274>
```



Conexão entre as interfaces

```
mininet> net
h1 h1-eth0:s1-eth1
h2 h2-eth0:s1-eth2
h3 h3-eth0:s2-eth1
h4 h4-eth0:s2-eth2
h5 h5-eth0:s3-eth1
h6 h6-eth0:s3-eth2
h7 h7-eth0:s4-eth1
h8 h8-eth0:s4-eth2
h9 h9-eth0:s4-eth3
s1 lo:  s1-eth1:h1-eth0 s1-eth2:h2-eth0 s1-eth3:s2-eth3
s2 lo:  s2-eth1:h3-eth0 s2-eth2:h4-eth0 s2-eth3:s1-eth3 s2-eth4:s3-eth3
s3 lo:  s3-eth1:h5-eth0 s3-eth2:h6-eth0 s3-eth3:s2-eth4 s3-eth4:s4-eth4
s4 lo:  s4-eth1:h7-eth0 s4-eth2:h8-eth0 s4-eth3:h9-eth0 s4-eth4:s3-eth4
```




Interfaces, endereços de IP e MAC

```
mininet> h1 ifconfig -a
h1-eth0  Link encap:Ethernet  HWaddr 00:00:00:00:00:01
         inet addr:10.0.0.1  Bcast:10.255.255.255  Mask:255.0.0.0
         UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
         RX packets:0 errors:0 dropped:0 overruns:0 frame:0
         TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

lo       Link encap:Local Loopback
         inet addr:127.0.0.1  Mask:255.0.0.0
         UP LOOPBACK RUNNING  MTU:65536  Metric:1
         RX packets:0 errors:0 dropped:0 overruns:0 frame:0
         TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1
         RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

Host h1 possui endereços IP e MAC terminados em 1



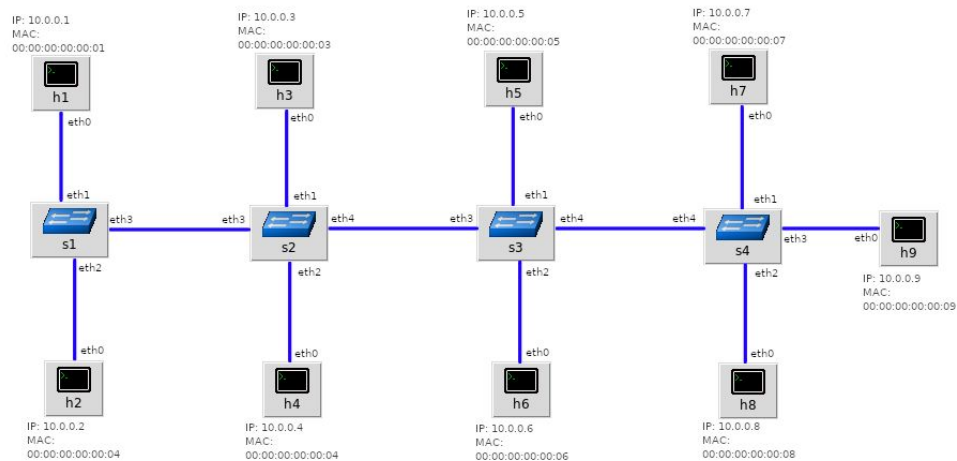
Interfaces, endereços de IP e MAC

```
mininet> h9 ifconfig -a
h9-eth0  Link encap:Ethernet  HWaddr 00:00:00:00:00:09
         inet addr:10.0.0.9  Bcast:10.255.255.255  Mask:255.0.0.0
         UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
         RX packets:0 errors:0 dropped:0 overruns:0 frame:0
         TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

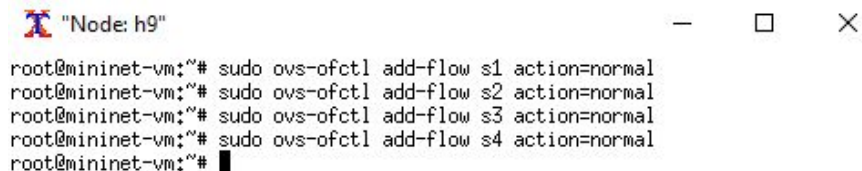
lo       Link encap:Local Loopback
         inet addr:127.0.0.1  Mask:255.0.0.0
         UP LOOPBACK RUNNING  MTU:65536  Metric:1
         RX packets:0 errors:0 dropped:0 overruns:0 frame:0
         TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1
         RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

Host h9 possui endereços IP e MAC terminados em 9

Ilustração da topologia



Configuração dos switches para operação normal



```
root@mininet-vm:~# sudo ovs-ofctl add-flow s1 action=normal
root@mininet-vm:~# sudo ovs-ofctl add-flow s2 action=normal
root@mininet-vm:~# sudo ovs-ofctl add-flow s3 action=normal
root@mininet-vm:~# sudo ovs-ofctl add-flow s4 action=normal
root@mininet-vm:~# █
```



Teste (operação normal do switch): ping geral

```
mininet> pingall
*** Ping: testing ping reachability
h1 -> h2 h3 h4 h5 h6 h7 h8 h9
h2 -> h1 h3 h4 h5 h6 h7 h8 h9
h3 -> h1 h2 h4 h5 h6 h7 h8 h9
h4 -> h1 h2 h3 h5 h6 h7 h8 h9
h5 -> h1 h2 h3 h4 h6 h7 h8 h9
h6 -> h1 h2 h3 h4 h5 h7 h8 h9
h7 -> h1 h2 h3 h4 h5 h6 h8 h9
h8 -> h1 h2 h3 h4 h5 h6 h7 h9
h9 -> h1 h2 h3 h4 h5 h6 h7 h8
*** Results: 0% dropped (72/72 received)
```



Teste (operação normal do switch): ping h1 para h9

```
mininet> h1 ping h9
PING 10.0.0.9 (10.0.0.9) 56(84) bytes of data.
64 bytes from 10.0.0.9: icmp_seq=1 ttl=64 time=1.10 ms
64 bytes from 10.0.0.9: icmp_seq=2 ttl=64 time=0.090 ms
64 bytes from 10.0.0.9: icmp_seq=3 ttl=64 time=0.053 ms
64 bytes from 10.0.0.9: icmp_seq=4 ttl=64 time=0.049 ms
64 bytes from 10.0.0.9: icmp_seq=5 ttl=64 time=0.039 ms
^C
--- 10.0.0.9 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4009ms
rtt min/avg/max/mdev = 0.039/0.267/1.108/0.420 ms
```



Teste (operação normal do switch): ping h2 para h8

```
mininet> h2 ping h8
PING 10.0.0.8 (10.0.0.8) 56(84) bytes of data.
64 bytes from 10.0.0.8: icmp_seq=1 ttl=64 time=4.56 ms
64 bytes from 10.0.0.8: icmp_seq=2 ttl=64 time=0.047 ms
64 bytes from 10.0.0.8: icmp_seq=3 ttl=64 time=0.043 ms
64 bytes from 10.0.0.8: icmp_seq=4 ttl=64 time=0.037 ms
64 bytes from 10.0.0.8: icmp_seq=5 ttl=64 time=0.045 ms
^C
--- 10.0.0.8 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4015ms
rtt min/avg/max/mdev = 0.037/0.947/4.564/1.808 ms
```



Regras baseadas em endereços MAC


- Configurar os switches com o OVS para permitir a comunicação entre hosts de forma seletiva, a partir do seu endereço MAC.
- De forma a cobrir hosts de todos os switches, mas evidenciar a implementação das regras, os nós que conseguem comunicar entre si são:
 - h1 -> h2, h3, h5, h7
 - h2 -> h1
 - h3 -> h1
 - h5 -> h1
 - h7 -> h1

Apagando regras anteriores e criando novas regras baseadas em endereços MAC

```
"Node: h9"
root@mininet-vm:~# sudo ovs-ofctl del-flows s1
root@mininet-vm:~# sudo ovs-ofctl del-flows s2
root@mininet-vm:~# sudo ovs-ofctl del-flows s3
root@mininet-vm:~# sudo ovs-ofctl del-flows s4
root@mininet-vm:~# sudo ovs-ofctl add-flow s1 dl_type=0x806,nw_proto=1,action=flood
root@mininet-vm:~# sudo ovs-ofctl add-flow s2 dl_type=0x806,nw_proto=1,action=flood

root@mininet-vm:~# sudo ovs-ofctl add-flow s3 dl_type=0x806,nw_proto=1,action=flood
root@mininet-vm:~# sudo ovs-ofctl add-flow s4 dl_type=0x806,nw_proto=1,action=flood
root@mininet-vm:~# █
```

Criando novas regras

 "Node: h9"

```
root@mininet-vm:~# sudo ovs-ofctl add-flow s1 dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:02,actions=output:2
root@mininet-vm:~# sudo ovs-ofctl add-flow s1 dl_src=00:00:00:00:00:02,dl_dst=00:00:00:00:00:01,actions=output:1
root@mininet-vm:~#
root@mininet-vm:~# sudo ovs-ofctl add-flow s1 dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:03,actions=output:3
root@mininet-vm:~# sudo ovs-ofctl add-flow s2 dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:03,actions=output:1
root@mininet-vm:~# sudo ovs-ofctl add-flow s1 dl_src=00:00:00:00:00:03,dl_dst=00:00:00:00:00:01,actions=output:1
root@mininet-vm:~# sudo ovs-ofctl add-flow s2 dl_src=00:00:00:00:00:03,dl_dst=00:00:00:00:00:01,actions=output:3
root@mininet-vm:~#
root@mininet-vm:~# sudo ovs-ofctl add-flow s1 dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:05,actions=output:3
root@mininet-vm:~# sudo ovs-ofctl add-flow s2 dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:05,actions=output:4
root@mininet-vm:~# sudo ovs-ofctl add-flow s3 dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:05,actions=output:1
root@mininet-vm:~# sudo ovs-ofctl add-flow s3 dl_src=00:00:00:00:00:05,dl_dst=00:00:00:00:00:01,actions=output:3
root@mininet-vm:~# sudo ovs-ofctl add-flow s2 dl_src=00:00:00:00:00:05,dl_dst=00:00:00:00:00:01,actions=output:3
root@mininet-vm:~# sudo ovs-ofctl add-flow s1 dl_src=00:00:00:00:00:05,dl_dst=00:00:00:00:00:01,actions=output:1
root@mininet-vm:~#
root@mininet-vm:~# sudo ovs-ofctl add-flow s1 dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:07,actions=output:3
root@mininet-vm:~# sudo ovs-ofctl add-flow s2 dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:07,actions=output:4
root@mininet-vm:~# sudo ovs-ofctl add-flow s3 dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:07,actions=output:4
root@mininet-vm:~# sudo ovs-ofctl add-flow s4 dl_src=00:00:00:00:00:01,dl_dst=00:00:00:00:00:07,actions=output:1
root@mininet-vm:~# sudo ovs-ofctl add-flow s4 dl_src=00:00:00:00:00:07,dl_dst=00:00:00:00:00:01,actions=output:4
root@mininet-vm:~# sudo ovs-ofctl add-flow s3 dl_src=00:00:00:00:00:07,dl_dst=00:00:00:00:00:01,actions=output:3
root@mininet-vm:~# sudo ovs-ofctl add-flow s2 dl_src=00:00:00:00:00:07,dl_dst=00:00:00:00:00:01,actions=output:3
root@mininet-vm:~# sudo ovs-ofctl add-flow s1 dl_src=00:00:00:00:00:07,dl_dst=00:00:00:00:00:01,actions=output:1
root@mininet-vm:~# █
```



Teste (operação por endereço MAC): ping geral

```
mininet> pingall
*** Ping: testing ping reachability
h1 -> h2 h3 X h5 X h7 X X
h2 -> h1 X X X X X X X
h3 -> h1 X X X X X X X
h4 -> X X X X X X X X
h5 -> h1 X X X X X X X
h6 -> X X X X X X X X
h7 -> h1 X X X X X X X
h8 -> X X X X X X X X
h9 -> X X X X X X X X
*** Results: 88% dropped (8/72 received)
```