Assignment01-MPTCP

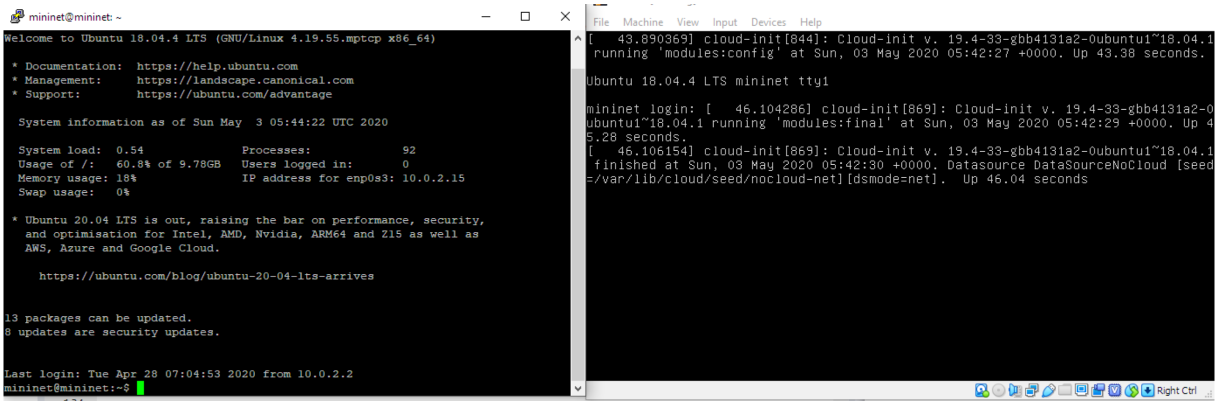
Zhenyu Yang 862187998

May 3, 2020

.

1 Mininet Setup

1.1 Configure the Virtualbox

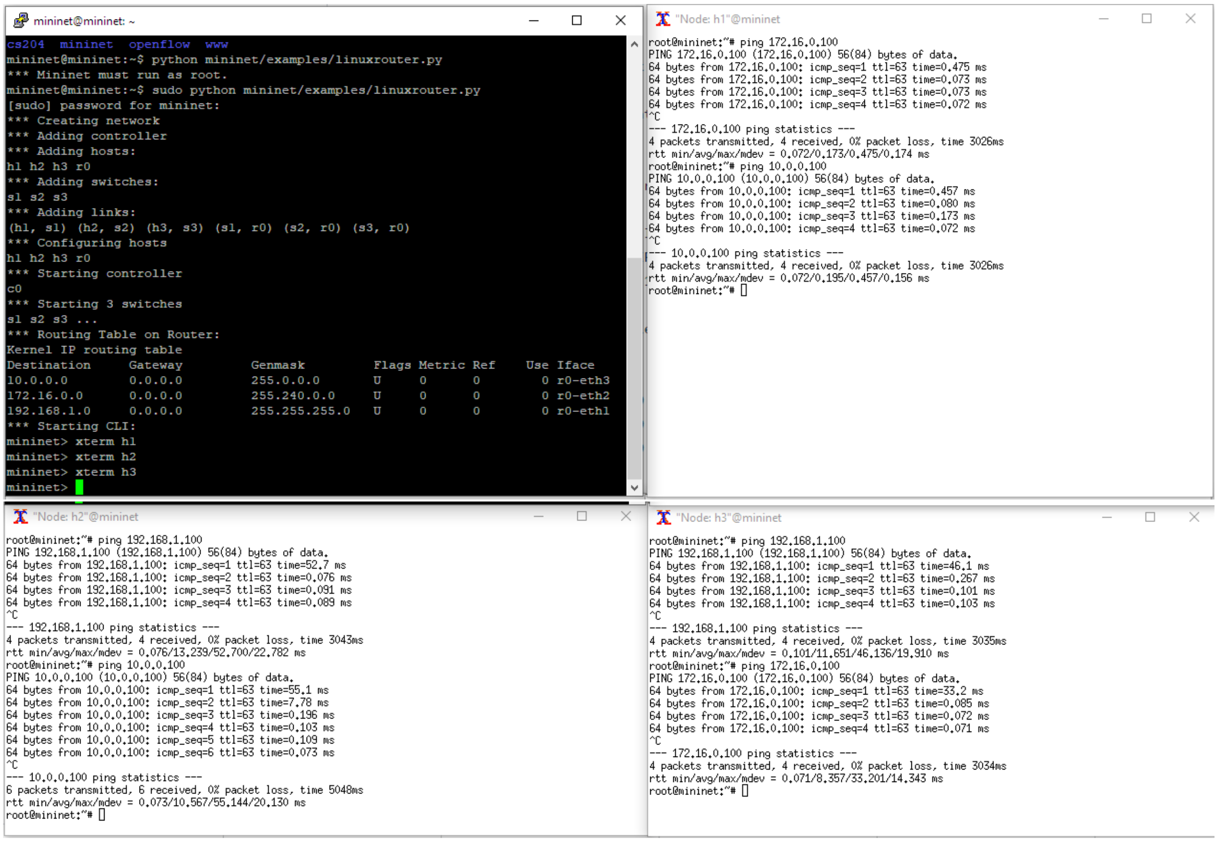


```
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.19.55.mptcp x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Sun May  3 05:44:22 UTC 2020

  System load:  0.54               Processes:             92
  Usage of /:   60.8% of 9.78GB   Users logged in:       0
  Memory usage: 18%                IP address for enp0s3: 10.0.2.15
  Swap usage:   0%

 * Ubuntu 20.04 LTS is out, raising the bar on performance, security,
   and optimisation for Intel, AMD, Nvidia, ARM64 and Z15 as well as
   AWS, Azure and Google Cloud.

     https://ubuntu.com/blog/ubuntu-20-04-lts-arrives


13 packages can be updated.
8 updates are security updates.


Last login: Tue Apr 28 07:04:53 2020 from 10.0.2.2
mininet@mininet:~$
```

```
[   43.890369] cloud-init[844]: Cloud-init v. 19.4-33-gbb4131a2-0ubuntu1~18.04.1
 running 'modules:config' at Sun, 03 May 2020 05:42:27 +0000. Up 43.38 seconds.

Ubuntu 18.04.4 LTS mininet tty1

mininet login: [   46.104286] cloud-init[869]: Cloud-init v. 19.4-33-gbb4131a2-0
ubuntu1~18.04.1 running 'modules:final' at Sun, 03 May 2020 05:42:29 +0000. Up 4
5.28 seconds.
[   46.106154] cloud-init[869]: Cloud-init v. 19.4-33-gbb4131a2-0ubuntu1~18.04.1
 finished at Sun, 03 May 2020 05:42:30 +0000. Datasource DataSourceNoCloud [seed
=/var/lib/cloud/seed/nocloud-net][dsmode=net].  Up 46.04 seconds
```

Figure 1: Configure the Virtualbox

1.2 Set up a sample topology



```
cs204  mininet  openflow  www
mininet@mininet:~$ python mininet/examples/linuxrouter.py
*** Mininet must run as root.
mininet@mininet:~$ sudo python mininet/examples/linuxrouter.py
[sudo] password for mininet:
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2 h3 r0
*** Adding switches:
s1 s2 s3
*** Adding links:
(h1, s1) (h2, s2) (h3, s3) (s1, r0) (s2, r0) (s3, r0)
*** Configuring hosts
h1 h2 h3 r0
*** Starting controller
c0
*** Starting 3 switches
s1 s2 s3 ...
*** Routing Table on Router:
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.0.0        0.0.0.0         255.0.0.0       U     0      0        0 r0-eth3
172.16.0.0      0.0.0.0         255.240.0.0     U     0      0        0 r0-eth2
192.168.1.0     0.0.0.0         255.255.255.0   U     0      0        0 r0-eth1
*** Starting CLI:
mininet> xterm h1
mininet> xterm h2
mininet> xterm h3
mininet> 
```

```
root@mininet:~# ping 172.16.0.100
PING 172.16.0.100 (172.16.0.100) 56(84) bytes of data.
64 bytes from 172.16.0.100: icmp_seq=1 ttl=63 time=0.475 ms
64 bytes from 172.16.0.100: icmp_seq=2 ttl=63 time=0.073 ms
64 bytes from 172.16.0.100: icmp_seq=3 ttl=63 time=0.073 ms
64 bytes from 172.16.0.100: icmp_seq=4 ttl=63 time=0.072 ms
^C
--- 172.16.0.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3026ms
rtt min/avg/max/mdev = 0.072/0.173/0.475/0.174 ms
root@mininet:~# ping 10.0.0.100
PING 10.0.0.100 (10.0.0.100) 56(84) bytes of data.
64 bytes from 10.0.0.100: icmp_seq=1 ttl=63 time=0.457 ms
64 bytes from 10.0.0.100: icmp_seq=2 ttl=63 time=0.080 ms
64 bytes from 10.0.0.100: icmp_seq=3 ttl=63 time=0.173 ms
64 bytes from 10.0.0.100: icmp_seq=4 ttl=63 time=0.072 ms
^C
--- 10.0.0.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3026ms
rtt min/avg/max/mdev = 0.072/0.195/0.457/0.156 ms
root@mininet:~# 
```

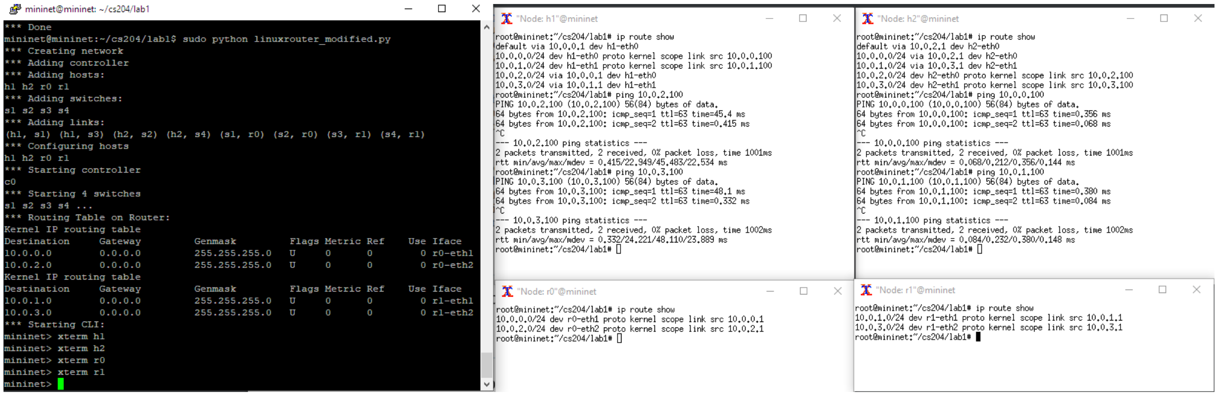
```
root@mininet:~# ping 192.168.1.100
PING 192.168.1.100 (192.168.1.100) 56(84) bytes of data.
64 bytes from 192.168.1.100: icmp_seq=1 ttl=63 time=52.7 ms
64 bytes from 192.168.1.100: icmp_seq=2 ttl=63 time=0.076 ms
64 bytes from 192.168.1.100: icmp_seq=3 ttl=63 time=0.091 ms
64 bytes from 192.168.1.100: icmp_seq=4 ttl=63 time=0.089 ms
^C
--- 192.168.1.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3043ms
rtt min/avg/max/mdev = 0.076/13.239/52.700/22.782 ms
root@mininet:~# ping 10.0.0.100
PING 10.0.0.100 (10.0.0.100) 56(84) bytes of data.
64 bytes from 10.0.0.100: icmp_seq=1 ttl=63 time=55.1 ms
64 bytes from 10.0.0.100: icmp_seq=2 ttl=63 time=7.78 ms
64 bytes from 10.0.0.100: icmp_seq=3 ttl=63 time=0.196 ms
64 bytes from 10.0.0.100: icmp_seq=4 ttl=63 time=0.103 ms
64 bytes from 10.0.0.100: icmp_seq=5 ttl=63 time=0.109 ms
64 bytes from 10.0.0.100: icmp_seq=6 ttl=63 time=0.073 ms
^C
--- 10.0.0.100 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5048ms
rtt min/avg/max/mdev = 0.073/10.567/55.144/20.130 ms
root@mininet:~# 
```

```
root@mininet:~# ping 192.168.1.100
PING 192.168.1.100 (192.168.1.100) 56(84) bytes of data.
64 bytes from 192.168.1.100: icmp_seq=1 ttl=63 time=46.1 ms
64 bytes from 192.168.1.100: icmp_seq=2 ttl=63 time=0.267 ms
64 bytes from 192.168.1.100: icmp_seq=3 ttl=63 time=0.101 ms
64 bytes from 192.168.1.100: icmp_seq=4 ttl=63 time=0.103 ms
^C
--- 192.168.1.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3035ms
rtt min/avg/max/mdev = 0.101/11.651/46.136/19.910 ms
root@mininet:~# ping 172.16.0.100
PING 172.16.0.100 (172.16.0.100) 56(84) bytes of data.
64 bytes from 172.16.0.100: icmp_seq=1 ttl=63 time=33.2 ms
64 bytes from 172.16.0.100: icmp_seq=2 ttl=63 time=0.085 ms
64 bytes from 172.16.0.100: icmp_seq=3 ttl=63 time=0.072 ms
64 bytes from 172.16.0.100: icmp_seq=4 ttl=63 time=0.071 ms
^C
--- 172.16.0.100 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3034ms
rtt min/avg/max/mdev = 0.071/8.357/33.201/14.343 ms
root@mininet:~# 
```

Figure 2: Set up a sample topology

h1,h2,h3 can ping each other.

1.3 Customize the topology



```
*** Done
mininet@mininet:~/cs204/lab1$ sudo python linuxrouter_modified.py
*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2 r0 r1
*** Adding switches:
s1 s2 s3 s4
*** Adding links:
(h1, s1) (h1, s3) (h2, s2) (h2, s4) (s1, r0) (s2, r0) (s3, r1) (s4, r1)
*** Configuring hosts
h1 h2 r0 r1
*** Starting controller
c0
*** Starting 4 switches
s1 s2 s3 s4 ...
*** Routing Table on Router:
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.0.0        0.0.0.0         255.255.255.0   U     0      0        0 r0-eth1
10.0.2.0        0.0.0.0         255.255.255.0   U     0      0        0 r0-eth2
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.1.0        0.0.0.0         255.255.255.0   U     0      0        0 r1-eth1
10.0.3.0        0.0.0.0         255.255.255.0   U     0      0        0 r1-eth2
*** Starting CLI:
mininet> xterm h1
mininet> xterm h2
mininet> xterm r0
mininet> xterm r1
mininet>
```

```
root@mininet:~/cs204/lab1# ip route show
default via 10.0.0.1 dev h1-eth0
10.0.0.0/24 dev h1-eth0 proto kernel scope link src 10.0.0.100
10.0.1.0/24 dev h1-eth1 proto kernel scope link src 10.0.1.100
10.0.2.0/24 via 10.0.0.1 dev h1-eth0
10.0.3.0/24 via 10.0.1.1 dev h1-eth1
root@mininet:~/cs204/lab1# ping 10.0.2.100
PING 10.0.2.100 (10.0.2.100) 56(84) bytes of data.
64 bytes from 10.0.2.100: icmp_seq=1 ttl=63 time=45.4 ms
64 bytes from 10.0.2.100: icmp_seq=2 ttl=63 time=0.415 ms
^C
--- 10.0.2.100 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.415/22.949/45.483/22.534 ms
root@mininet:~/cs204/lab1# ping 10.0.3.100
PING 10.0.3.100 (10.0.3.100) 56(84) bytes of data.
64 bytes from 10.0.3.100: icmp_seq=1 ttl=63 time=48.1 ms
64 bytes from 10.0.3.100: icmp_seq=2 ttl=63 time=0.332 ms
^C
--- 10.0.3.100 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 0.332/24.221/48.110/23.889 ms
root@mininet:~/cs204/lab1#
```

```
root@mininet:~/cs204/lab1# ip route show
default via 10.0.2.1 dev h2-eth0
10.0.0.0/24 via 10.0.2.1 dev h2-eth0
10.0.1.0/24 via 10.0.3.1 dev h2-eth1
10.0.2.0/24 dev h2-eth0 proto kernel scope link src 10.0.2.100
10.0.3.0/24 dev h2-eth1 proto kernel scope link src 10.0.3.100
root@mininet:~/cs204/lab1# ping 10.0.0.100
PING 10.0.0.100 (10.0.0.100) 56(84) bytes of data.
64 bytes from 10.0.0.100: icmp_seq=1 ttl=63 time=0.356 ms
64 bytes from 10.0.0.100: icmp_seq=2 ttl=63 time=0.068 ms
^C
--- 10.0.0.100 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.068/0.212/0.356/0.144 ms
root@mininet:~/cs204/lab1# ping 10.0.1.100
PING 10.0.1.100 (10.0.1.100) 56(84) bytes of data.
64 bytes from 10.0.1.100: icmp_seq=1 ttl=63 time=0.380 ms
64 bytes from 10.0.1.100: icmp_seq=2 ttl=63 time=0.084 ms
^C
--- 10.0.1.100 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 0.084/0.232/0.380/0.148 ms
root@mininet:~/cs204/lab1#
```

```
root@mininet:~/cs204/lab1# ip route show
10.0.0.0/24 dev r0-eth1 proto kernel scope link src 10.0.0.1
10.0.2.0/24 dev r0-eth2 proto kernel scope link src 10.0.2.1
root@mininet:~/cs204/lab1#
```

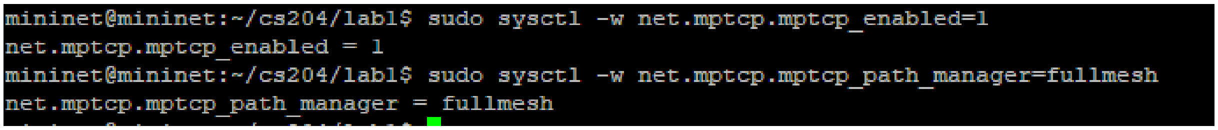
```
root@mininet:~/cs204/lab1# ip route show
10.0.1.0/24 dev r1-eth1 proto kernel scope link src 10.0.1.1
10.0.3.0/24 dev r1-eth2 proto kernel scope link src 10.0.3.1
root@mininet:~/cs204/lab1#
```

Figure 3: Customize the topology

Both paths can work now!

2 MPTCP Setup

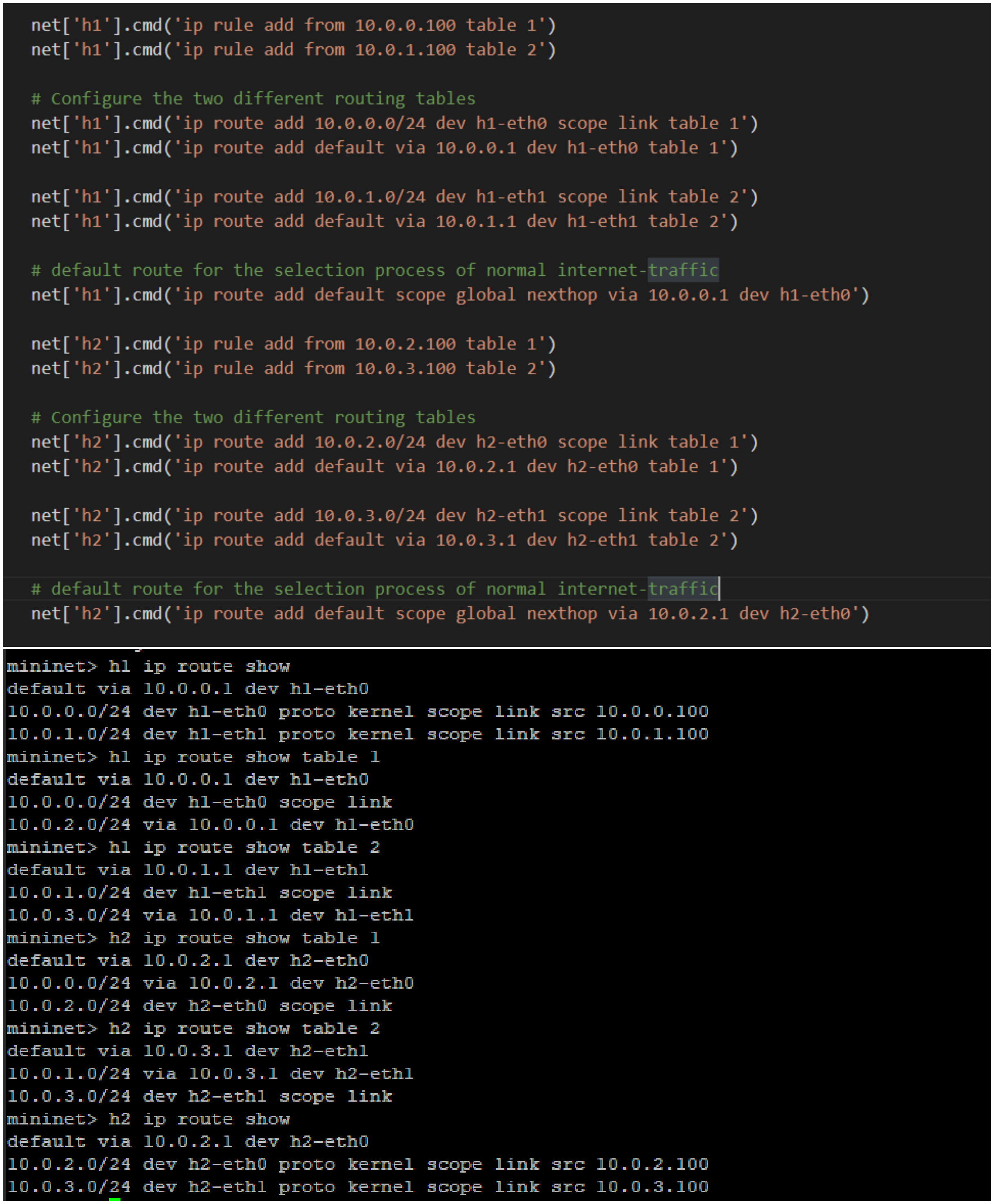
2.1 Enable MPTCP



```
mininet@mininet:~/cs204/lab1$ sudo sysctl -w net.mptcp.mptcp_enabled=1
net.mptcp.mptcp_enabled = 1
mininet@mininet:~/cs204/lab1$ sudo sysctl -w net.mptcp.mptcp_path_manager=fullmesh
net.mptcp.mptcp_path_manager = fullmesh
mininet@mininet:~/cs204/lab1$
```

Figure 4: Enable MPTCP

2.2 Configure the MPTCP routing tables



```
net['h1'].cmd('ip rule add from 10.0.0.100 table 1')
net['h1'].cmd('ip rule add from 10.0.1.100 table 2')

# Configure the two different routing tables
net['h1'].cmd('ip route add 10.0.0.0/24 dev h1-eth0 scope link table 1')
net['h1'].cmd('ip route add default via 10.0.0.1 dev h1-eth0 table 1')

net['h1'].cmd('ip route add 10.0.1.0/24 dev h1-eth1 scope link table 2')
net['h1'].cmd('ip route add default via 10.0.1.1 dev h1-eth1 table 2')

# default route for the selection process of normal internet-traffic
net['h1'].cmd('ip route add default scope global nexthop via 10.0.0.1 dev h1-eth0')

net['h2'].cmd('ip rule add from 10.0.2.100 table 1')
net['h2'].cmd('ip rule add from 10.0.3.100 table 2')

# Configure the two different routing tables
net['h2'].cmd('ip route add 10.0.2.0/24 dev h2-eth0 scope link table 1')
net['h2'].cmd('ip route add default via 10.0.2.1 dev h2-eth0 table 1')

net['h2'].cmd('ip route add 10.0.3.0/24 dev h2-eth1 scope link table 2')
net['h2'].cmd('ip route add default via 10.0.3.1 dev h2-eth1 table 2')

# default route for the selection process of normal internet-traffic
net['h2'].cmd('ip route add default scope global nexthop via 10.0.2.1 dev h2-eth0')
```

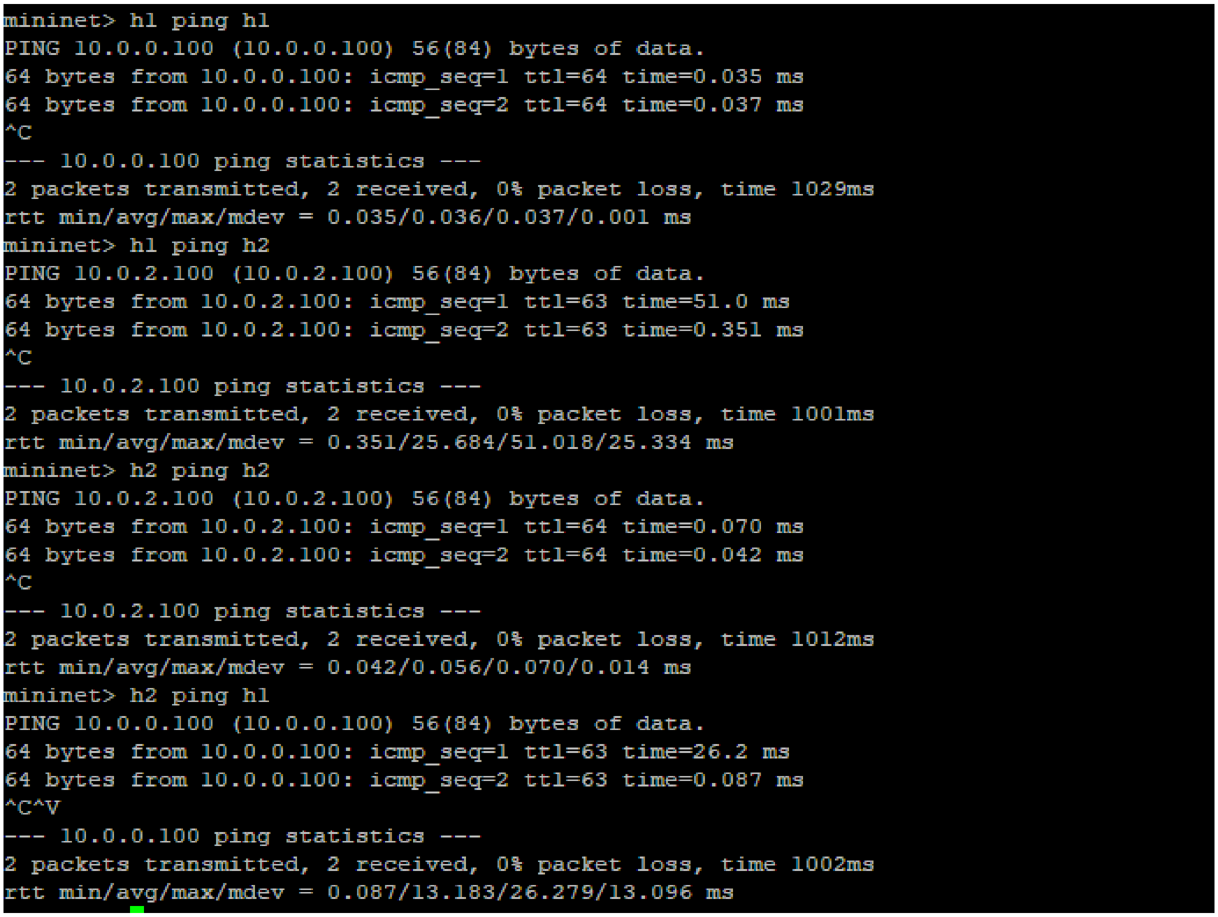
```
mininet> h1 ip route show
default via 10.0.0.1 dev h1-eth0
10.0.0.0/24 dev h1-eth0 proto kernel scope link src 10.0.0.100
10.0.1.0/24 dev h1-eth1 proto kernel scope link src 10.0.1.100
mininet> h1 ip route show table 1
default via 10.0.0.1 dev h1-eth0
10.0.0.0/24 dev h1-eth0 scope link
10.0.2.0/24 via 10.0.0.1 dev h1-eth0
mininet> h1 ip route show table 2
default via 10.0.1.1 dev h1-eth1
10.0.1.0/24 dev h1-eth1 scope link
10.0.3.0/24 via 10.0.1.1 dev h1-eth1
mininet> h2 ip route show table 1
default via 10.0.2.1 dev h2-eth0
10.0.0.0/24 via 10.0.2.1 dev h2-eth0
10.0.2.0/24 dev h2-eth0 scope link
mininet> h2 ip route show table 2
default via 10.0.3.1 dev h2-eth1
10.0.1.0/24 via 10.0.3.1 dev h2-eth1
10.0.3.0/24 dev h2-eth1 scope link
mininet> h2 ip route show
default via 10.0.2.1 dev h2-eth0
10.0.2.0/24 dev h2-eth0 proto kernel scope link src 10.0.2.100
10.0.3.0/24 dev h2-eth1 proto kernel scope link src 10.0.3.100
```

Figure 5: Configure the MPTCP routing tables

2.3 Verification

2.3.1 Ping tests

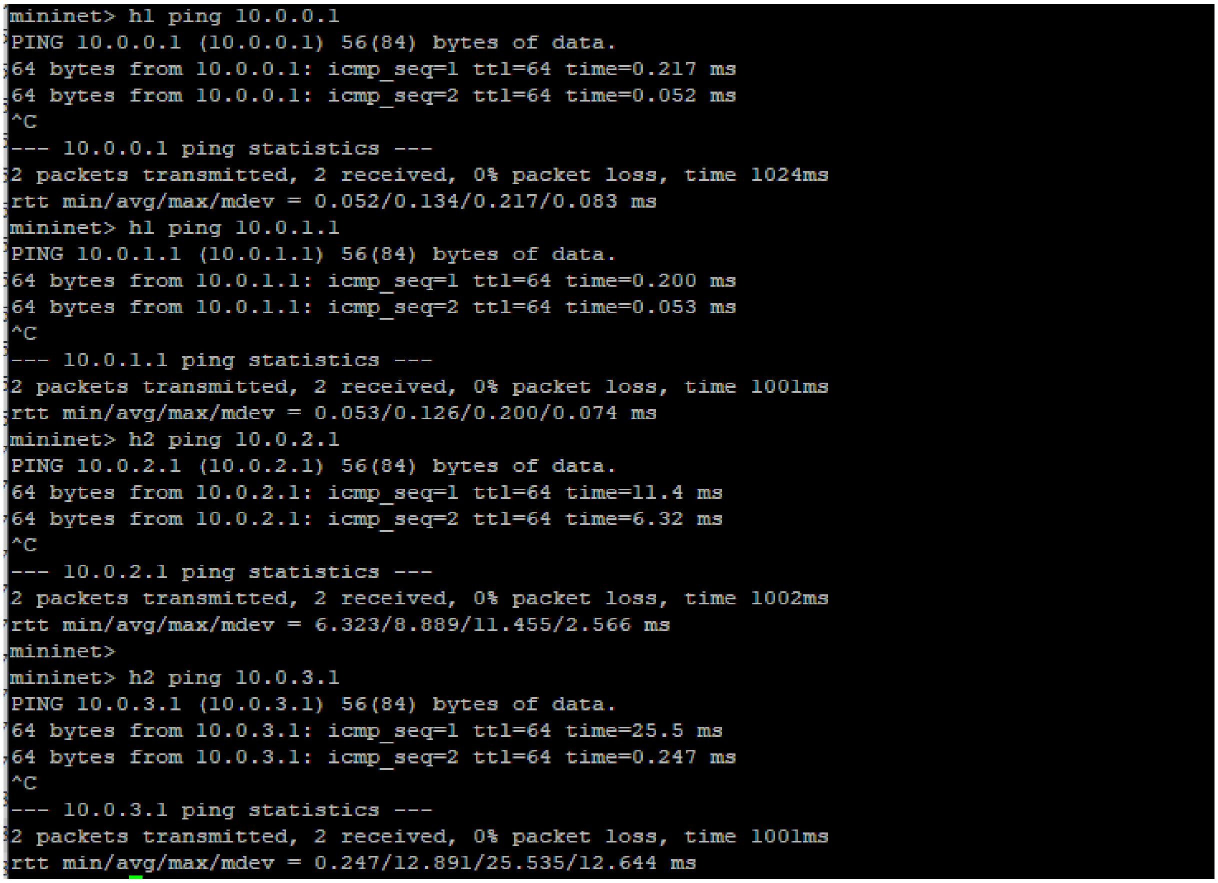
(1). ping from host to host (Verified)



```
mininet> h1 ping h1
PING 10.0.0.100 (10.0.0.100) 56(84) bytes of data.
64 bytes from 10.0.0.100: icmp_seq=1 ttl=64 time=0.035 ms
64 bytes from 10.0.0.100: icmp_seq=2 ttl=64 time=0.037 ms
^C
--- 10.0.0.100 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1029ms
rtt min/avg/max/mdev = 0.035/0.036/0.037/0.001 ms
mininet> h1 ping h2
PING 10.0.2.100 (10.0.2.100) 56(84) bytes of data.
64 bytes from 10.0.2.100: icmp_seq=1 ttl=63 time=51.0 ms
64 bytes from 10.0.2.100: icmp_seq=2 ttl=63 time=0.351 ms
^C
--- 10.0.2.100 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.351/25.684/51.018/25.334 ms
mininet> h2 ping h2
PING 10.0.2.100 (10.0.2.100) 56(84) bytes of data.
64 bytes from 10.0.2.100: icmp_seq=1 ttl=64 time=0.070 ms
64 bytes from 10.0.2.100: icmp_seq=2 ttl=64 time=0.042 ms
^C
--- 10.0.2.100 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1012ms
rtt min/avg/max/mdev = 0.042/0.056/0.070/0.014 ms
mininet> h2 ping h1
PING 10.0.0.100 (10.0.0.100) 56(84) bytes of data.
64 bytes from 10.0.0.100: icmp_seq=1 ttl=63 time=26.2 ms
64 bytes from 10.0.0.100: icmp_seq=2 ttl=63 time=0.087 ms
^C^V
--- 10.0.0.100 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 0.087/13.183/26.279/13.096 ms
```

Figure 6: ping from host to host

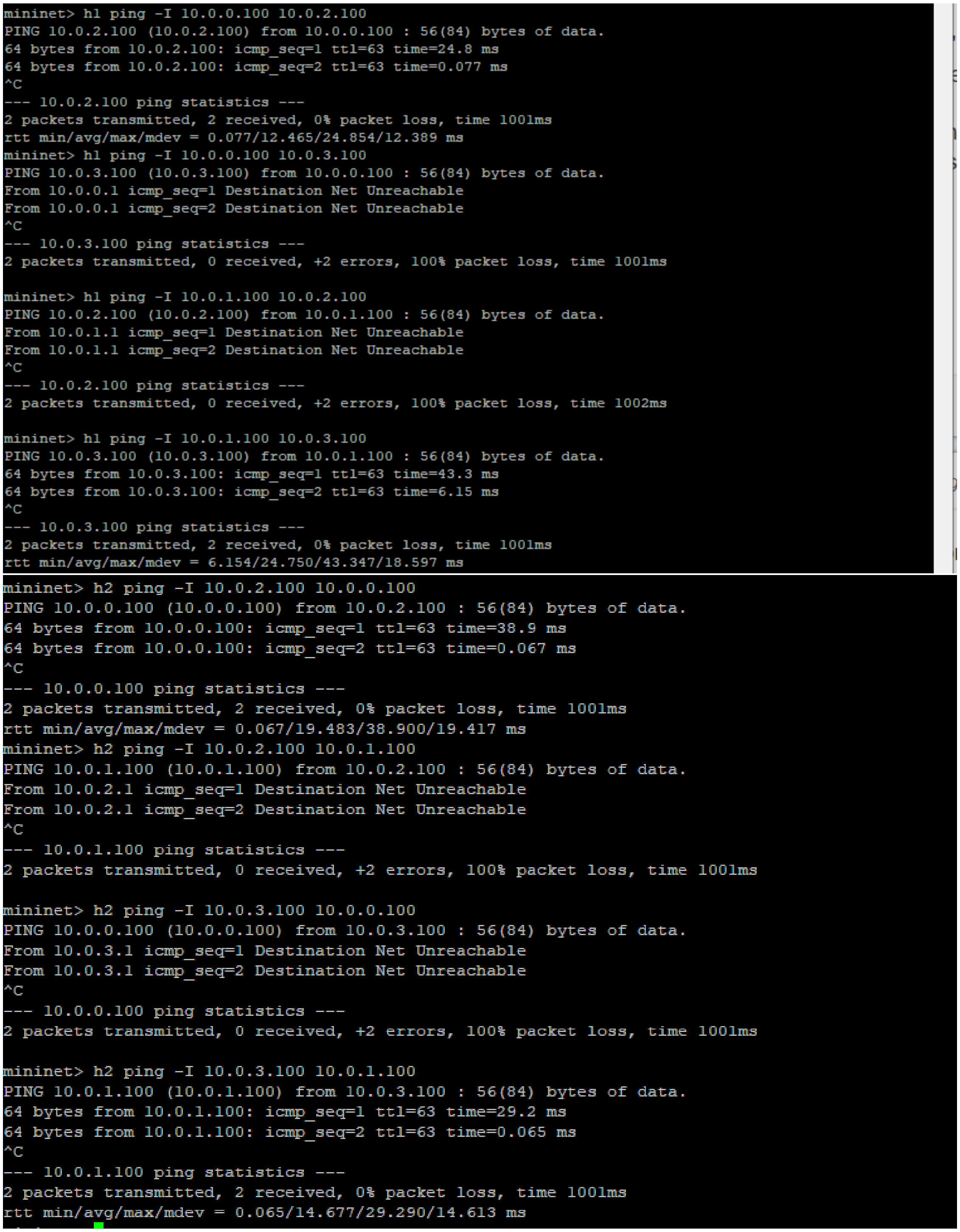
(2). ping from host to gateway (Verified)



```
mininet> h1 ping 10.0.0.1
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=0.217 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.052 ms
^C
--- 10.0.0.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1024ms
rtt min/avg/max/mdev = 0.052/0.134/0.217/0.083 ms
mininet> h1 ping 10.0.1.1
PING 10.0.1.1 (10.0.1.1) 56(84) bytes of data.
64 bytes from 10.0.1.1: icmp_seq=1 ttl=64 time=0.200 ms
64 bytes from 10.0.1.1: icmp_seq=2 ttl=64 time=0.053 ms
^C
--- 10.0.1.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.053/0.126/0.200/0.074 ms
mininet> h2 ping 10.0.2.1
PING 10.0.2.1 (10.0.2.1) 56(84) bytes of data.
64 bytes from 10.0.2.1: icmp_seq=1 ttl=64 time=11.4 ms
64 bytes from 10.0.2.1: icmp_seq=2 ttl=64 time=6.32 ms
^C
--- 10.0.2.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 6.323/8.889/11.455/2.566 ms
mininet>
mininet> h2 ping 10.0.3.1
PING 10.0.3.1 (10.0.3.1) 56(84) bytes of data.
64 bytes from 10.0.3.1: icmp_seq=1 ttl=64 time=25.5 ms
64 bytes from 10.0.3.1: icmp_seq=2 ttl=64 time=0.247 ms
^C
--- 10.0.3.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.247/12.891/25.535/12.644 ms
```

Figure 7: ping from host to gateway

(3). ping from interface to interface (Verified)



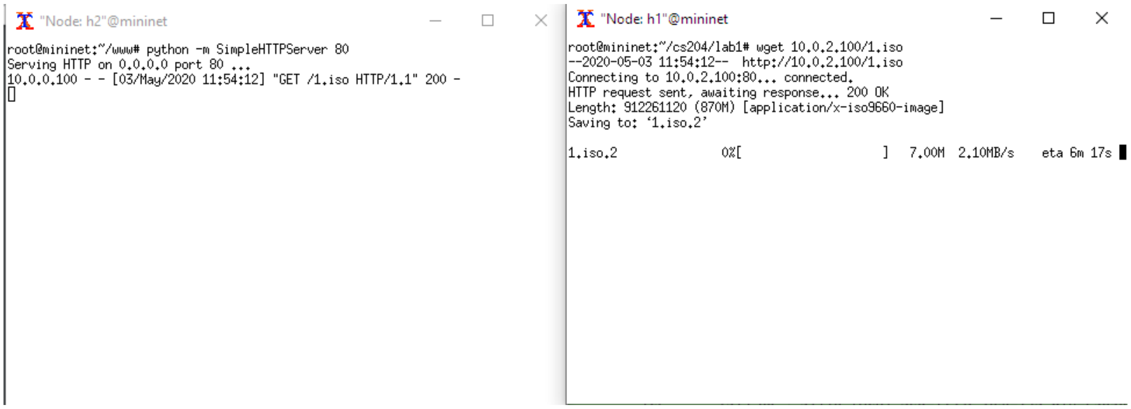
```
mininet> h1 ping -I 10.0.0.100 10.0.2.100
PING 10.0.2.100 (10.0.2.100) from 10.0.0.100 : 56(84) bytes of data.
64 bytes from 10.0.2.100: icmp_seq=1 ttl=63 time=24.8 ms
64 bytes from 10.0.2.100: icmp_seq=2 ttl=63 time=0.077 ms
^C
--- 10.0.2.100 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.077/12.465/24.854/12.389 ms
mininet> h1 ping -I 10.0.0.100 10.0.3.100
PING 10.0.3.100 (10.0.3.100) from 10.0.0.100 : 56(84) bytes of data.
From 10.0.0.1 icmp_seq=1 Destination Net Unreachable
From 10.0.0.1 icmp_seq=2 Destination Net Unreachable
^C
--- 10.0.3.100 ping statistics ---
2 packets transmitted, 0 received, +2 errors, 100% packet loss, time 1001ms

mininet> h1 ping -I 10.0.1.100 10.0.2.100
PING 10.0.2.100 (10.0.2.100) from 10.0.1.100 : 56(84) bytes of data.
From 10.0.1.1 icmp_seq=1 Destination Net Unreachable
From 10.0.1.1 icmp_seq=2 Destination Net Unreachable
^C
--- 10.0.2.100 ping statistics ---
2 packets transmitted, 0 received, +2 errors, 100% packet loss, time 1002ms

mininet> h1 ping -I 10.0.1.100 10.0.3.100
PING 10.0.3.100 (10.0.3.100) from 10.0.1.100 : 56(84) bytes of data.
64 bytes from 10.0.3.100: icmp_seq=1 ttl=63 time=43.3 ms
64 bytes from 10.0.3.100: icmp_seq=2 ttl=63 time=6.15 ms
^C
--- 10.0.3.100 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 6.154/24.750/43.347/18.597 ms
```

```
mininet> h2 ping -I 10.0.2.100 10.0.0.100
PING 10.0.0.100 (10.0.0.100) from 10.0.2.100 : 56(84) bytes of data.
64 bytes from 10.0.0.100: icmp_seq=1 ttl=63 time=38.9 ms
64 bytes from 10.0.0.100: icmp_seq=2 ttl=63 time=0.067 ms
^C
--- 10.0.0.100 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.067/19.483/38.900/19.417 ms
mininet> h2 ping -I 10.0.2.100 10.0.1.100
PING 10.0.1.100 (10.0.1.100) from 10.0.2.100 : 56(84) bytes of data.
From 10.0.2.1 icmp_seq=1 Destination Net Unreachable
From 10.0.2.1 icmp_seq=2 Destination Net Unreachable
^C
--- 10.0.1.100 ping statistics ---
2 packets transmitted, 0 received, +2 errors, 100% packet loss, time 1001ms

mininet> h2 ping -I 10.0.3.100 10.0.0.100
PING 10.0.0.100 (10.0.0.100) from 10.0.3.100 : 56(84) bytes of data.
From 10.0.3.1 icmp_seq=1 Destination Net Unreachable
From 10.0.3.1 icmp_seq=2 Destination Net Unreachable
^C
--- 10.0.0.100 ping statistics ---
2 packets transmitted, 0 received, +2 errors, 100% packet loss, time 1001ms

mininet> h2 ping -I 10.0.3.100 10.0.1.100
PING 10.0.1.100 (10.0.1.100) from 10.0.3.100 : 56(84) bytes of data.
64 bytes from 10.0.1.100: icmp_seq=1 ttl=63 time=29.2 ms
64 bytes from 10.0.1.100: icmp_seq=2 ttl=63 time=0.065 ms
^C
--- 10.0.1.100 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.065/14.677/29.290/14.613 ms
```

Figure 8: ping from interface to interface

2.3.2 Download tests

(1). Set up a web server on h2

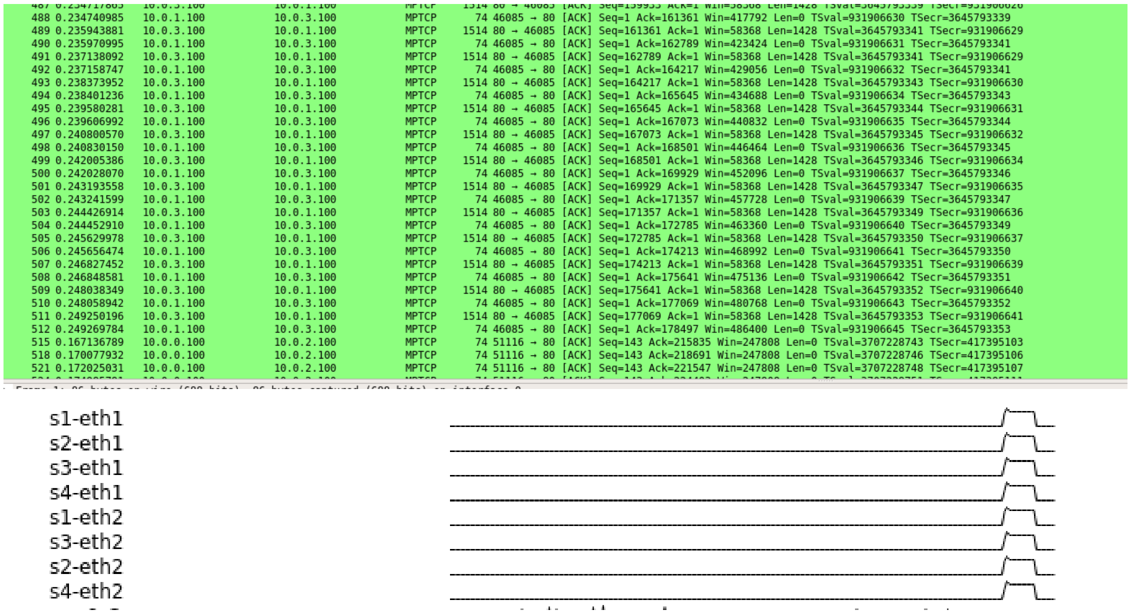


```
root@mininet:~/www# python -m SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
10.0.0.100 - - [03/May/2020 11:54:12] "GET /1.iso HTTP/1.1" 200 -
```

```
root@mininet:~/cs204/lab1# wget 10.0.2.100/1.iso
--2020-05-03 11:54:12--  http://10.0.2.100/1.iso
Connecting to 10.0.2.100:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 912261120 (870M) [application/x-iso9660-image]
Saving to: ‘1.iso.2’

1.iso.2              0%[                    ]   7.00M  2.10MB/s    eta 6m 17s
```

Figure 9: web server

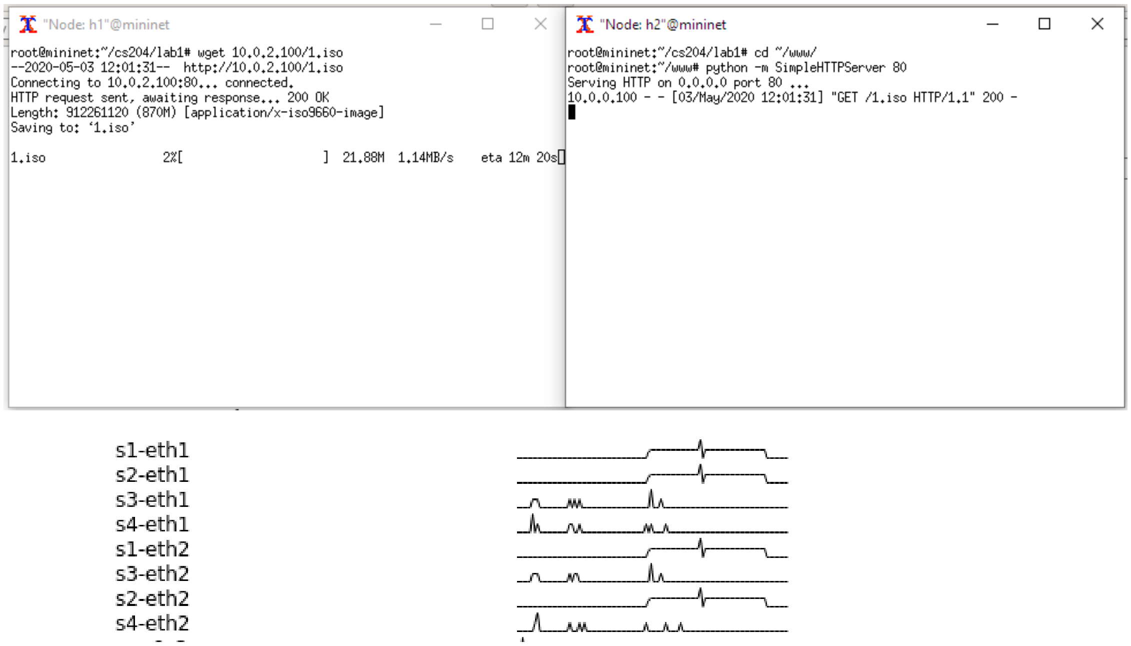
(2). MPTCP results



```
487 0.234717863   10.0.3.100    10.0.1.100    MPTCP   1514 80 → 46085 [ACK] Seq=159933 Ack=1 Win=58368 Len=1428 TSval=3645793339 TSecr=931906628
488 0.234740985   10.0.1.100    10.0.3.100    MPTCP     74 46085 → 80 [ACK] Seq=1 Ack=161361 Win=417792 Len=0 TSval=931906630 TSecr=3645793339
489 0.235943881   10.0.3.100    10.0.1.100    MPTCP   1514 80 → 46085 [ACK] Seq=161361 Ack=1 Win=58368 Len=1428 TSval=3645793341 TSecr=931906629
490 0.235970995   10.0.1.100    10.0.3.100    MPTCP     74 46085 → 80 [ACK] Seq=1 Ack=162789 Win=423424 Len=0 TSval=931906631 TSecr=3645793341
491 0.237138092   10.0.3.100    10.0.1.100    MPTCP   1514 80 → 46085 [ACK] Seq=162789 Ack=1 Win=58368 Len=1428 TSval=3645793341 TSecr=931906629
492 0.237158747   10.0.1.100    10.0.3.100    MPTCP     74 46085 → 80 [ACK] Seq=1 Ack=164217 Win=429056 Len=0 TSval=931906632 TSecr=3645793341
493 0.238373952   10.0.3.100    10.0.1.100    MPTCP   1514 80 → 46085 [ACK] Seq=164217 Ack=1 Win=58368 Len=1428 TSval=3645793343 TSecr=931906630
494 0.238401236   10.0.1.100    10.0.3.100    MPTCP     74 46085 → 80 [ACK] Seq=1 Ack=165645 Win=434688 Len=0 TSval=931906634 TSecr=3645793343
495 0.239580281   10.0.3.100    10.0.1.100    MPTCP   1514 80 → 46085 [ACK] Seq=165645 Ack=1 Win=58368 Len=1428 TSval=3645793344 TSecr=931906631
496 0.239606992   10.0.1.100    10.0.3.100    MPTCP     74 46085 → 80 [ACK] Seq=1 Ack=167073 Win=440832 Len=0 TSval=931906635 TSecr=3645793344
497 0.240800570   10.0.3.100    10.0.1.100    MPTCP   1514 80 → 46085 [ACK] Seq=167073 Ack=1 Win=58368 Len=1428 TSval=3645793345 TSecr=931906632
498 0.240830150   10.0.1.100    10.0.3.100    MPTCP     74 46085 → 80 [ACK] Seq=1 Ack=168501 Win=446464 Len=0 TSval=931906636 TSecr=3645793345
499 0.242005386   10.0.3.100    10.0.1.100    MPTCP   1514 80 → 46085 [ACK] Seq=168501 Ack=1 Win=58368 Len=1428 TSval=3645793346 TSecr=931906634
500 0.242028070   10.0.1.100    10.0.3.100    MPTCP     74 46085 → 80 [ACK] Seq=1 Ack=169929 Win=452096 Len=0 TSval=931906637 TSecr=3645793346
501 0.243193558   10.0.3.100    10.0.1.100    MPTCP   1514 80 → 46085 [ACK] Seq=169929 Ack=1 Win=58368 Len=1428 TSval=3645793347 TSecr=931906635
502 0.243241599   10.0.1.100    10.0.3.100    MPTCP     74 46085 → 80 [ACK] Seq=1 Ack=171357 Win=457728 Len=0 TSval=931906639 TSecr=3645793347
503 0.244426914   10.0.3.100    10.0.1.100    MPTCP   1514 80 → 46085 [ACK] Seq=171357 Ack=1 Win=58368 Len=1428 TSval=3645793349 TSecr=931906636
504 0.244452910   10.0.1.100    10.0.3.100    MPTCP     74 46085 → 80 [ACK] Seq=1 Ack=172785 Win=463360 Len=0 TSval=931906640 TSecr=3645793349
505 0.245629978   10.0.3.100    10.0.1.100    MPTCP   1514 80 → 46085 [ACK] Seq=172785 Ack=1 Win=58368 Len=1428 TSval=3645793350 TSecr=931906637
506 0.245656474   10.0.1.100    10.0.3.100    MPTCP     74 46085 → 80 [ACK] Seq=1 Ack=174213 Win=468992 Len=0 TSval=931906641 TSecr=3645793350
507 0.246827452   10.0.3.100    10.0.1.100    MPTCP   1514 80 → 46085 [ACK] Seq=174213 Ack=1 Win=58368 Len=1428 TSval=3645793351 TSecr=931906639
508 0.246848581   10.0.1.100    10.0.3.100    MPTCP     74 46085 → 80 [ACK] Seq=1 Ack=175641 Win=475136 Len=0 TSval=931906642 TSecr=3645793351
509 0.248038349   10.0.3.100    10.0.1.100    MPTCP   1514 80 → 46085 [ACK] Seq=175641 Ack=1 Win=58368 Len=1428 TSval=3645793352 TSecr=931906640
510 0.248058942   10.0.1.100    10.0.3.100    MPTCP     74 46085 → 80 [ACK] Seq=1 Ack=177069 Win=480768 Len=0 TSval=931906643 TSecr=3645793352
511 0.249250196   10.0.3.100    10.0.1.100    MPTCP   1514 80 → 46085 [ACK] Seq=177069 Ack=1 Win=58368 Len=1428 TSval=3645793353 TSecr=931906641
512 0.249269784   10.0.1.100    10.0.3.100    MPTCP     74 46085 → 80 [ACK] Seq=1 Ack=178497 Win=486400 Len=0 TSval=931906645 TSecr=3645793353
515 0.167136789   10.0.0.100    10.0.2.100    MPTCP     74 51116 → 80 [ACK] Seq=143 Ack=215835 Win=247808 Len=0 TSval=3707228743 TSecr=417395103
518 0.170077932   10.0.0.100    10.0.2.100    MPTCP     74 51116 → 80 [ACK] Seq=143 Ack=218691 Win=247808 Len=0 TSval=3707228746 TSecr=417395106
521 0.172025031   10.0.0.100    10.0.2.100    MPTCP     74 51116 → 80 [ACK] Seq=143 Ack=221547 Win=247808 Len=0 TSval=3707228748 TSecr=417395107
Frame 1: 86 bytes on wire (688 bits), 86 bytes captured (688 bits) on interface 0
```

Figure 10: MPTCP results

(2). Normal TCP results



```
root@mininet:~/cs204/lab1# wget 10.0.2.100/1.iso
--2020-05-03 12:01:31--  http://10.0.2.100/1.iso
Connecting to 10.0.2.100:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 912261120 (870M) [application/x-iso9660-image]
Saving to: ‘1.iso’

1.iso               2%[                    ]  21.88M  1.14MB/s    eta 12m 20s
```

```
root@mininet:~/cs204/lab1# cd ~/www/
root@mininet:~/www# python -m SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
10.0.0.100 - - [03/May/2020 12:01:31] "GET /1.iso HTTP/1.1" 200 -
```

Figure 11: results

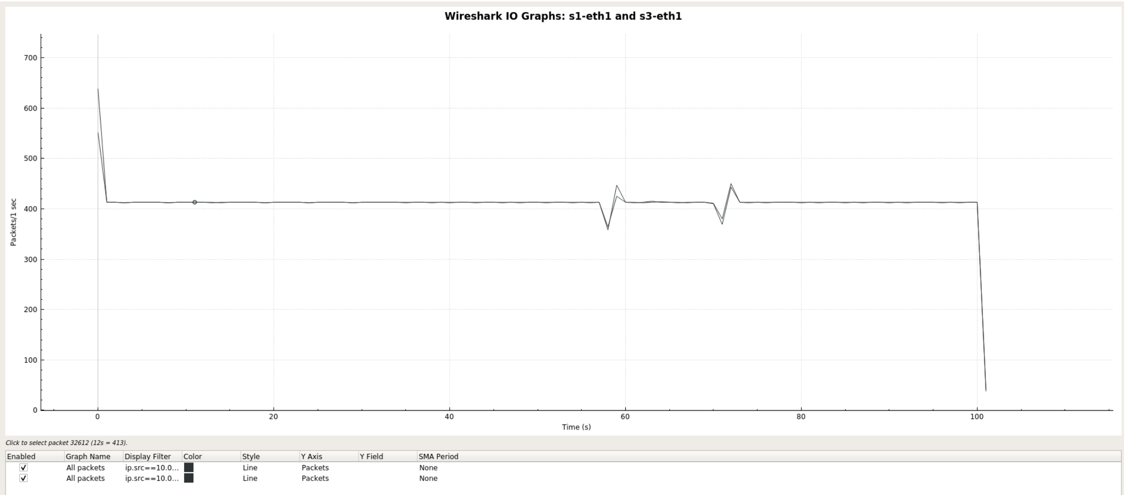
From the results we can conclude that MPTCP is actually worked, it improves the download speed(1.14mb/s to 2.10mb/s) when the both links are **”full loaded”**.

3 Report Questions

3.1 file transfer

Because the wireshark running extremely slow when capture a large amount packets, so I only download 15% of the iso file

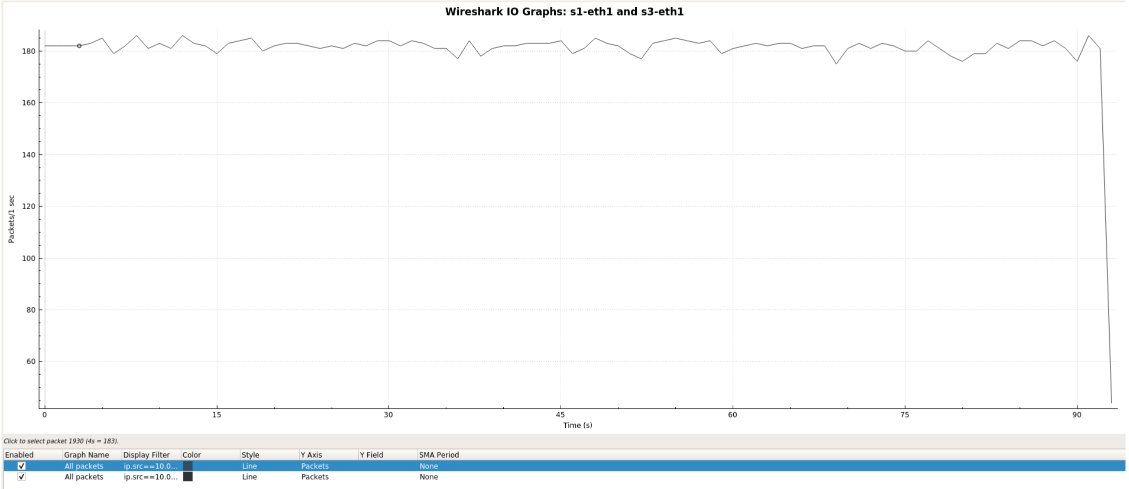
Here are the result plot for **MPTCP** transfer:



Click to select packet 32612 (12s = 413).

Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period
☑	All packets	ip.src==10.0...	■	Line	Packets		None
☑	All packets	ip.src==10.0...	■	Line	Packets		None

Here are the result plot for **Normal Tcp** transfer:



Click to select packet 1930 (4s = 183).

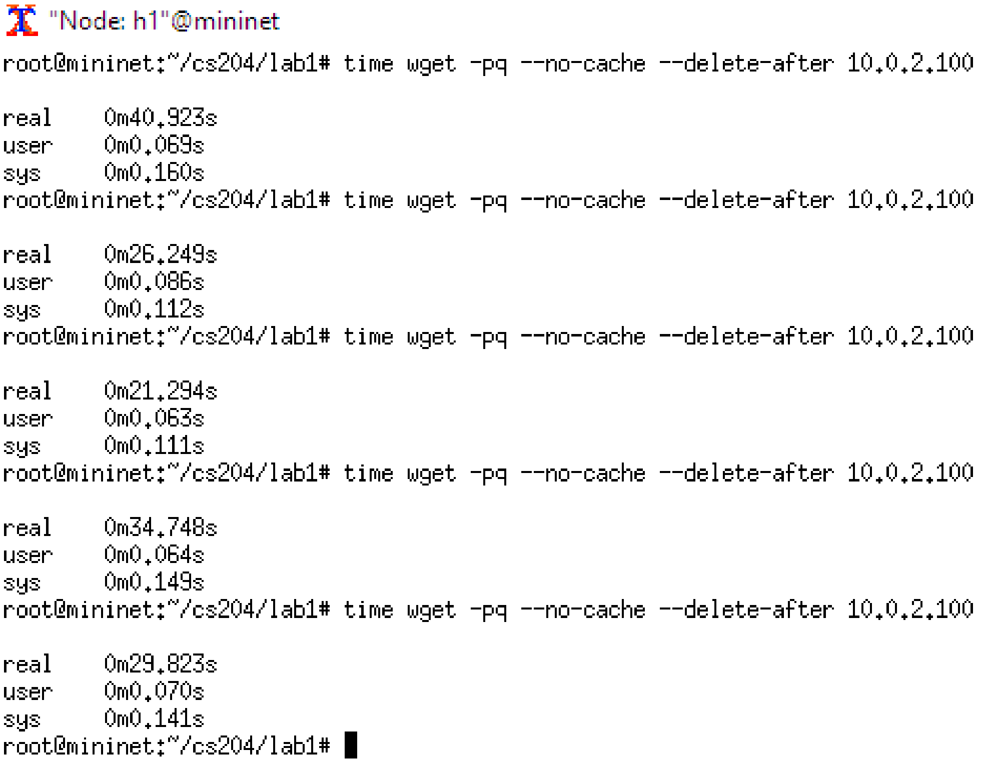
Enabled	Graph Name	Display Filter	Color	Style	Y Axis	Y Field	SMA Period
☑	All packets	ip.src==10.0...	■	Line	Packets		None
☑	All packets	ip.src==10.0...	■	Line	Packets		None

Comparing the two plots we know that MPTCP indeed using two paths to average the workload, and obtains double speed as normal TCP's.

MPTCP increase the total throughput and also decrease the total download time. The reason why it has so much improvement is that MPTCP perfectly used another link to transfer data and single tcp link cannot reach the maximum speed due to the 10mbits/s limitation of outgoing rate.

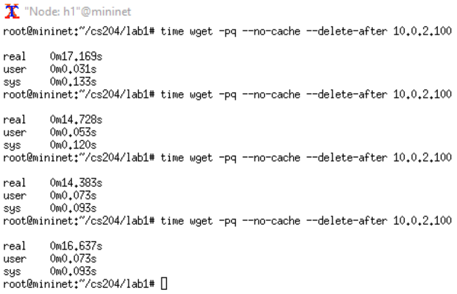
3.2 web browsing

Here are the result **wget** time for **MPTCP** browsing:

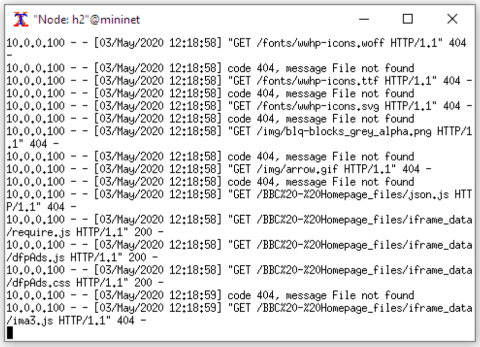


```
root@mininet:~/cs204/lab1# time wget -pq --no-cache --delete-after 10.0.2.100

real    0m40.923s
user    0m0.069s
sys     0m0.160s
root@mininet:~/cs204/lab1# time wget -pq --no-cache --delete-after 10.0.2.100

real    0m26.249s
user    0m0.086s
sys     0m0.112s
root@mininet:~/cs204/lab1# time wget -pq --no-cache --delete-after 10.0.2.100

real    0m21.294s
user    0m0.063s
sys     0m0.111s
root@mininet:~/cs204/lab1# time wget -pq --no-cache --delete-after 10.0.2.100

real    0m34.748s
user    0m0.064s
sys     0m0.149s
root@mininet:~/cs204/lab1# time wget -pq --no-cache --delete-after 10.0.2.100

real    0m29.823s
user    0m0.070s
sys     0m0.141s
root@mininet:~/cs204/lab1# █
```

Here are the result **wget** time for **Normal Tcp** browsing:
Comparing the two results we know that MPTCP actually slow down the web browsing speed.



```
root@mininet:~/cs204/lab1# time wget -pq --no-cache --delete-after 10.0.2.100

real    0m17.169s
user    0m0.031s
sys     0m0.133s
root@mininet:~/cs204/lab1# time wget -pq --no-cache --delete-after 10.0.2.100

real    0m14.728s
user    0m0.053s
sys     0m0.120s
root@mininet:~/cs204/lab1# time wget -pq --no-cache --delete-after 10.0.2.100

real    0m14.383s
user    0m0.073s
sys     0m0.093s
root@mininet:~/cs204/lab1# time wget -pq --no-cache --delete-after 10.0.2.100

real    0m16.637s
user    0m0.073s
sys     0m0.093s
root@mininet:~/cs204/lab1# []
```

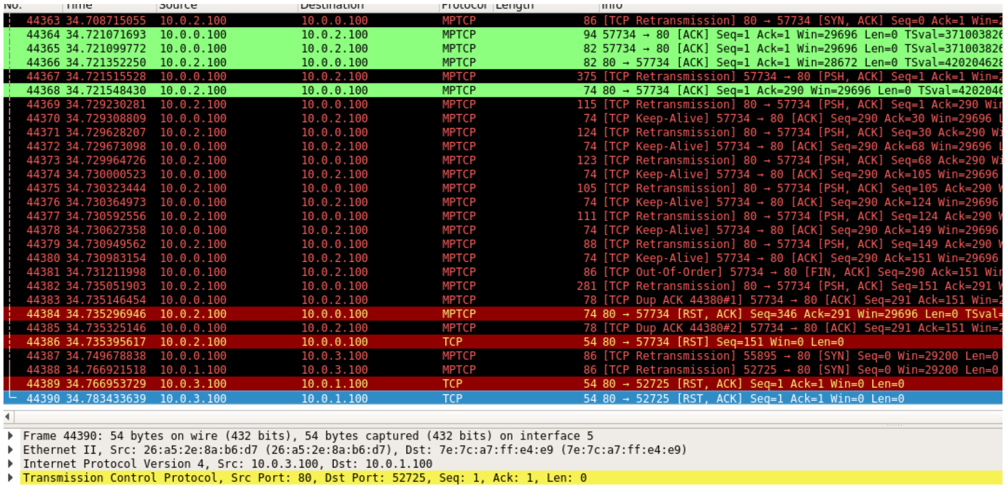


```
10.0.0.100 - - [03/May/2020 12:18:58] "GET /fonts/wwhp-icons.woff HTTP/1.1" 404
-
10.0.0.100 - - [03/May/2020 12:18:58] code 404, message File not found
10.0.0.100 - - [03/May/2020 12:18:58] "GET /fonts/wwhp-icons.ttf HTTP/1.1" 404 -
10.0.0.100 - - [03/May/2020 12:18:58] code 404, message File not found
10.0.0.100 - - [03/May/2020 12:18:58] "GET /fonts/wwhp-icons.svg HTTP/1.1" 404 -
10.0.0.100 - - [03/May/2020 12:18:58] code 404, message File not found
10.0.0.100 - - [03/May/2020 12:18:58] "GET /img/blq-blocks_grey_alpha.png HTTP/1
.1" 404 -
10.0.0.100 - - [03/May/2020 12:18:58] code 404, message File not found
10.0.0.100 - - [03/May/2020 12:18:58] "GET /img/arrow.gif HTTP/1.1" 404 -
10.0.0.100 - - [03/May/2020 12:18:58] code 404, message File not found
10.0.0.100 - - [03/May/2020 12:18:58] "GET /BBC%20-%20Homepage_files/json.js HTT
P/1.1" 404 -
10.0.0.100 - - [03/May/2020 12:18:58] "GET /BBC%20-%20Homepage_files/iframe_data
/require.js HTTP/1.1" 200 -
10.0.0.100 - - [03/May/2020 12:18:58] "GET /BBC%20-%20Homepage_files/iframe_data
/dfpAds.js HTTP/1.1" 200 -
10.0.0.100 - - [03/May/2020 12:18:58] "GET /BBC%20-%20Homepage_files/iframe_data
/dfpAds.css HTTP/1.1" 200 -
10.0.0.100 - - [03/May/2020 12:18:59] code 404, message File not found
10.0.0.100 - - [03/May/2020 12:18:59] "GET /BBC%20-%20Homepage_files/iframe_data
/ima3.js HTTP/1.1" 404 -
█
```

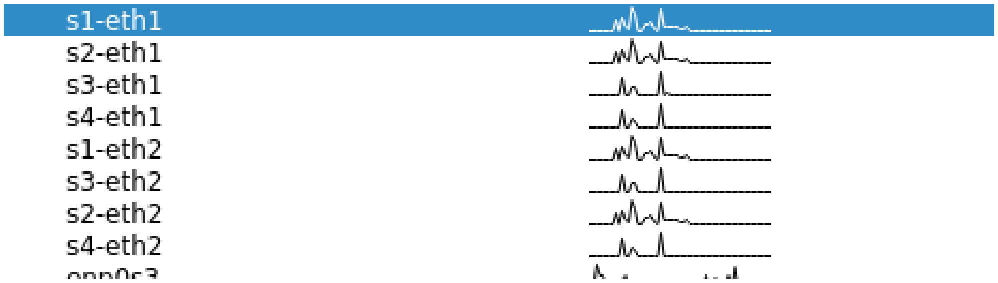
Here is the table only for ”GET” request:

	Wget time	Fraction of objects on path 1
MPTCP enabled	30	100%
MPTCP disabled	15	100%

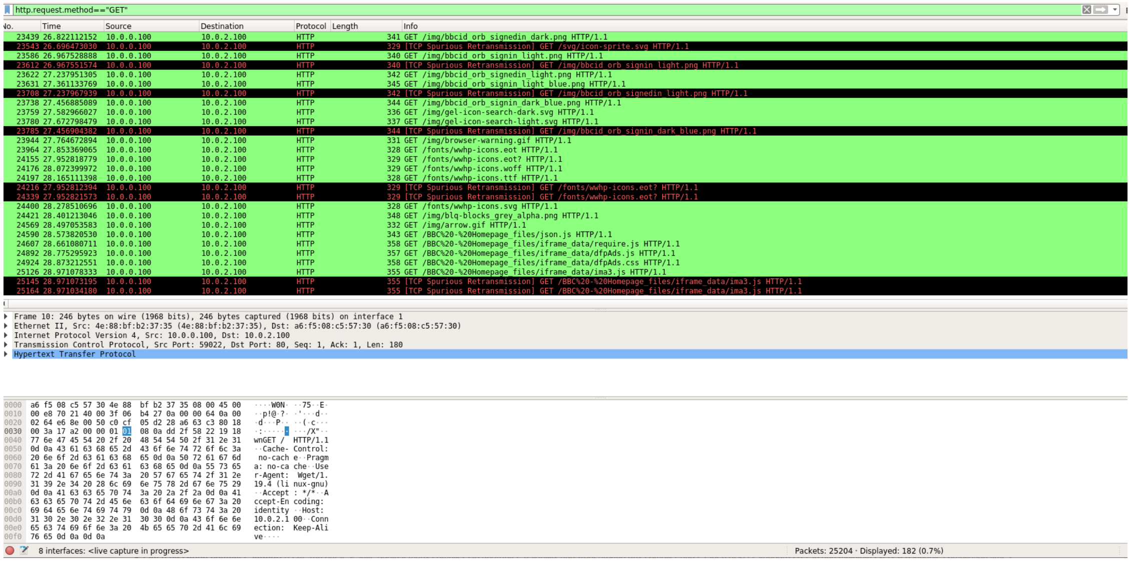
From the result table, MPTCP does not help to reduce the page load time, almost 100% **GET request** just from the upper path in both MPTCP enabled and disable situation. The general packets capture results when MPTCP enabled are shown as follow:



```
No.   Time         Source       Destination  Protocol Length Info
 44363 34.708715055 10.0.2.100   10.0.0.100   MPTCP      86 [TCP Retransmission] 80 → 57734 [SYN, ACK] Seq=0 Ack=1 Win=2
 44364 34.721071693 10.0.0.100   10.0.2.100   MPTCP      94 57734 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0 TSval=371003826
 44365 34.721099772 10.0.0.100   10.0.2.100   MPTCP      82 57734 → 80 [ACK] Seq=1 Ack=1 Win=29696 Len=0 TSval=371003826
 44366 34.721352250 10.0.2.100   10.0.0.100   MPTCP      82 80 → 57734 [ACK] Seq=1 Ack=1 Win=28672 Len=0 TSval=420204628
 44367 34.721515528 10.0.0.100   10.0.2.100   MPTCP     375 [TCP Retransmission] 57734 → 80 [PSH, ACK] Seq=1 Ack=1 Win=2
 44368 34.721548430 10.0.2.100   10.0.0.100   MPTCP      74 80 → 57734 [ACK] Seq=1 Ack=290 Win=29696 Len=0 TSval=4202046
 44369 34.729230281 10.0.2.100   10.0.0.100   MPTCP     115 [TCP Retransmission] 80 → 57734 [PSH, ACK] Seq=1 Ack=290 Wir
 44370 34.729308809 10.0.0.100   10.0.2.100   MPTCP      74 [TCP Keep-Alive] 57734 → 80 [ACK] Seq=290 Ack=30 Win=29696 L
 44371 34.729628207 10.0.2.100   10.0.0.100   MPTCP     124 [TCP Retransmission] 80 → 57734 [PSH, ACK] Seq=30 Ack=290 Wi
 44372 34.729673098 10.0.0.100   10.0.2.100   MPTCP      74 [TCP Keep-Alive] 57734 → 80 [ACK] Seq=290 Ack=68 Win=29696 L
 44373 34.729964726 10.0.2.100   10.0.0.100   MPTCP     123 [TCP Retransmission] 80 → 57734 [PSH, ACK] Seq=68 Ack=290 Wi
 44374 34.730000523 10.0.0.100   10.0.2.100   MPTCP      74 [TCP Keep-Alive] 57734 → 80 [ACK] Seq=290 Ack=105 Win=29696
 44375 34.730323444 10.0.2.100   10.0.0.100   MPTCP     105 [TCP Retransmission] 80 → 57734 [PSH, ACK] Seq=105 Ack=290 W
 44376 34.730364973 10.0.0.100   10.0.2.100   MPTCP      74 [TCP Keep-Alive] 57734 → 80 [ACK] Seq=290 Ack=124 Win=29696
 44377 34.730592556 10.0.2.100   10.0.0.100   MPTCP     111 [TCP Retransmission] 80 → 57734 [PSH, ACK] Seq=124 Ack=290 W
 44378 34.730627358 10.0.0.100   10.0.2.100   MPTCP      74 [TCP Keep-Alive] 57734 → 80 [ACK] Seq=290 Ack=149 Win=29696
 44379 34.730949562 10.0.2.100   10.0.0.100   MPTCP      88 [TCP Retransmission] 80 → 57734 [PSH, ACK] Seq=149 Ack=290 W
 44380 34.730983154 10.0.0.100   10.0.2.100   MPTCP      74 [TCP Keep-Alive] 57734 → 80 [ACK] Seq=290 Ack=151 Win=29696
 44381 34.731211998 10.0.0.100   10.0.2.100   MPTCP      86 [TCP Out-Of-Order] 57734 → 80 [FIN, ACK] Seq=290 Ack=151 Wir
 44382 34.735051903 10.0.2.100   10.0.0.100   MPTCP     281 [TCP Retransmission] 80 → 57734 [PSH, ACK] Seq=151 Ack=291 W
 44383 34.735146454 10.0.0.100   10.0.2.100   MPTCP      78 [TCP Dup ACK 44380#1] 57734 → 80 [ACK] Seq=291 Ack=151 Win=2
 44384 34.735296946 10.0.2.100   10.0.0.100   MPTCP      74 80 → 57734 [RST, ACK] Seq=346 Ack=291 Win=29696 Len=0 TSval=
 44385 34.735325146 10.0.0.100   10.0.2.100   MPTCP      78 [TCP Dup ACK 44380#2] 57734 → 80 [ACK] Seq=291 Ack=151 Win=2
 44386 34.735395617 10.0.2.100   10.0.0.100   TCP        54 80 → 57734 [RST] Seq=151 Win=0 Len=0
 44387 34.749678838 10.0.0.100   10.0.3.100   MPTCP      86 [TCP Retransmission] 55895 → 80 [SYN] Seq=0 Win=29200 Len=0
 44388 34.766921518 10.0.1.100   10.0.3.100   MPTCP      86 [TCP Retransmission] 52725 → 80 [SYN] Seq=0 Win=29200 Len=0
 44389 34.766953729 10.0.3.100   10.0.1.100   TCP        54 80 → 52725 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
 44390 34.783433639 10.0.3.100   10.0.1.100   TCP        54 80 → 52725 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0

▸ Frame 44390: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 5
▸ Ethernet II, Src: 26:a5:2e:8a:b6:d7 (26:a5:2e:8a:b6:d7), Dst: 7e:7c:a7:ff:e4:e9 (7e:7c:a7:ff:e4:e9)
▸ Internet Protocol Version 4, Src: 10.0.3.100, Dst: 10.0.1.100
▸ Transmission Control Protocol, Src Port: 80, Dst Port: 52725, Seq: 1, Ack: 1, Len: 0
```



There are many **MPTCP keepalive** and **Retransmission** messages, and from the traffic we also find that at most of time the **throughput** is in a **low level**. Comparing with MPTCP enabled, the **MPTCP disabled** has better efficiency:



```
http.request.method=="GET"

No.   Time         Source       Destination  Protocol Length Info
 23439 26.822112152 10.0.0.100   10.0.2.100   HTTP      341 GET /img/bbcid_orb_signedin_dark.png HTTP/1.1
 23543 26.696473030 10.0.0.100   10.0.2.100   HTTP      329 [TCP Spurious Retransmission] GET /svg/icon-sprite.svg HTTP/1.1
 23586 26.967528888 10.0.0.100   10.0.2.100   HTTP      340 GET /img/bbcid_orb_signin_light.png HTTP/1.1
 23612 26.967551574 10.0.0.100   10.0.2.100   HTTP      340 [TCP Spurious Retransmission] GET /img/bbcid_orb_signin_light.png HTTP/1.1
 23622 27.237951305 10.0.0.100   10.0.2.100   HTTP      342 GET /img/bbcid_orb_signedin_light.png HTTP/1.1
 23631 27.361133769 10.0.0.100   10.0.2.100   HTTP      345 GET /img/bbcid_orb_signin_light_blue.png HTTP/1.1
 23708 27.237967939 10.0.0.100   10.0.2.100   HTTP      342 [TCP Spurious Retransmission] GET /img/bbcid_orb_signedin_light.png HTTP/1.1
 23738 27.456885089 10.0.0.100   10.0.2.100   HTTP      344 GET /img/bbcid_orb_signin_dark_blue.png HTTP/1.1
 23759 27.582966027 10.0.0.100   10.0.2.100   HTTP      336 GET /img/gel-icon-search-dark.svg HTTP/1.1
 23780 27.672798479 10.0.0.100   10.0.2.100   HTTP      337 GET /img/gel-icon-search-light.svg HTTP/1.1
 23785 27.456904382 10.0.0.100   10.0.2.100   HTTP      344 [TCP Spurious Retransmission] GET /img/bbcid_orb_signin_dark_blue.png HTTP/1.1
 23944 27.764672894 10.0.0.100   10.0.2.100   HTTP      331 GET /img/browser-warning.gif HTTP/1.1
 23964 27.853369065 10.0.0.100   10.0.2.100   HTTP      328 GET /fonts/wwhp-icons.eot HTTP/1.1
 24155 27.952818779 10.0.0.100   10.0.2.100   HTTP      329 GET /fonts/wwhp-icons.eot? HTTP/1.1
 24176 28.072399972 10.0.0.100   10.0.2.100   HTTP      329 GET /fonts/wwhp-icons.woff HTTP/1.1
 24197 28.165111398 10.0.0.100   10.0.2.100   HTTP      328 GET /fonts/wwhp-icons.ttf HTTP/1.1
 24216 27.952812394 10.0.0.100   10.0.2.100   HTTP      329 [TCP Spurious Retransmission] GET /fonts/wwhp-icons.eot? HTTP/1.1
 24339 27.952821573 10.0.0.100   10.0.2.100   HTTP      329 [TCP Spurious Retransmission] GET /fonts/wwhp-icons.eot? HTTP/1.1
 24400 28.278510696 10.0.0.100   10.0.2.100   HTTP      328 GET /fonts/wwhp-icons.svg HTTP/1.1
 24421 28.401213046 10.0.0.100   10.0.2.100   HTTP      348 GET /img/blq-blocks_grey_alpha.png HTTP/1.1
 24569 28.497053583 10.0.0.100   10.0.2.100   HTTP      332 GET /img/arrow.gif HTTP/1.1
 24590 28.573820530 10.0.0.100   10.0.2.100   HTTP      343 GET /BBC%20-%20Homepage_files/json.js HTTP/1.1
 24607 28.661080711 10.0.0.100   10.0.2.100   HTTP      358 GET /BBC%20-%20Homepage_files/iframe_data/require.js HTTP/1.1
 24892 28.775295923 10.0.0.100   10.0.2.100   HTTP      357 GET /BBC%20-%20Homepage_files/iframe_data/dfpAds.js HTTP/1.1
 24924 28.873212551 10.0.0.100   10.0.2.100   HTTP      358 GET /BBC%20-%20Homepage_files/iframe_data/dfpAds.css HTTP/1.1
 25126 28.971078333 10.0.0.100   10.0.2.100   HTTP      355 GET /BBC%20-%20Homepage_files/iframe_data/ima3.js HTTP/1.1
 25145 28.971073195 10.0.0.100   10.0.2.100   HTTP      355 [TCP Spurious Retransmission] GET /BBC%20-%20Homepage_files/iframe_data/ima3.js HTTP/1.1
 25164 28.971034180 10.0.0.100   10.0.2.100   HTTP      355 [TCP Spurious Retransmission] GET /BBC%20-%20Homepage_files/iframe_data/ima3.js HTTP/1.1

▸ Frame 10: 246 bytes on wire (1968 bits), 246 bytes captured (1968 bits) on interface 1
▸ Ethernet II, Src: 4e:88:bf:b2:37:35 (4e:88:bf:b2:37:35), Dst: a6:f5:08:c5:57:30 (a6:f5:08:c5:57:30)
▸ Internet Protocol Version 4, Src: 10.0.0.100, Dst: 10.0.2.100
▸ Transmission Control Protocol, Src Port: 59022, Dst Port: 80, Seq: 1, Ack: 1, Len: 180
▸ Hypertext Transfer Protocol

0000  a6 f5 08 c5 57 30 4e 88  bf b2 37 35 08 00 45 00   ····W0N· ··75··E·
0010  00 e8 70 21 40 00 3f 06  b4 27 0a 00 00 64 0a 00   ··p!@·?· ·'···d··
0020  02 64 e6 8e 00 50 c0 cf  05 d2 28 a6 63 c3 80 18   ·d···P·· ··(·c···
0030  00 3a 17 a2 00 00 01 01  08 0a dd 2f 58 22 19 18   ·:······ ···/X"··
0040  77 6e 47 45 54 20 2f 20  48 54 54 50 2f 31 2e 31   wnGET /  HTTP/1.1
0050  0d 0a 43 61 63 68 65 2d  43 6f 6e 74 72 6f 6c 3a   ··Cache- Control:
0060  20 6e 6f 2d 63 61 63 68  65 0d 0a 50 72 61 67 6d    no-cach e··Pragm
0070  61 3a 20 6e 6f 2d 63 61  63 68 65 0d 0a 55 73 65   a: no-ca che··Use
0080  72 2d 41 67 65 6e 74 3a  20 57 67 65 74 2f 31 2e   r-Agent:  Wget/1.
0090  31 39 2e 34 20 28 6c 69  6e 75 78 2d 67 6e 75 29   19.4 (li nux-gnu)
00a0  0d 0a 41 63 63 65 70 74  3a 20 2a 2f 2a 0d 0a 41   ··Accept : */*··A
00b0  63 63 65 70 74 2d 45 6e  63 6f 64 69 6e 67 3a 20   ccept-En coding: 
00c0  69 64 65 6e 74 69 74 79  0d 0a 48 6f 73 74 3a 20   identity ··Host: 
00d0  31 30 2e 30 2e 32 2e 31  30 30 0d 0a 43 6f 6e 6e   10.0.2.1 00··Conn
00e0  65 63 74 69 6f 6e 3a 20  4b 65 65 70 2d 41 6c 69   ection:  Keep-Ali
00f0  76 65 0d 0a 0d 0a                                  ve····

8 interfaces: <live capture in progress>        Packets: 25204 · Displayed: 182 (0.7%)
```

I think **wget** is not a reasonable approximation of a web browse because it has to keep tcp link alive. I think a better browse should just download those static files.