

Lab3-Report

本文档是对lab3所有task和question的总结，所有截图分别位于对应task的文件夹下。

Task 1 请在你自己的环境中完成上面的连通性测试，并以截图的形式分别记录Node:h1和Node:h2中iperf的输出结果。要求：截图尽可能清晰，且需要连同任务栏的系统时间一起截图，后续的所有截图都要求包含系统时间。

Node:h1中 iperf的输出结果：

A terminal window titled "Node: h1" with a system clock at the top showing "3月 21 10:53". The terminal shows a user at the prompt "root@taoyucheng-dev:/home/taoyucheng#" attempting to run "ipconfig" and "ifconfig". Both commands are not found. The user then runs "ifconfig", which displays details for the "h1-eth0" interface (10.0.0.1) and the "lo" loopback interface (127.0.0.1). Finally, the user runs "iperf -s", which starts a server listening on TCP port 5001. A client connects from 10.0.0.2 port 42814, and the test results show a transfer of 44.2 GBytes at a bandwidth of 38.0 Gbits/sec over a 10-second interval.

```
3月 21 10:53
"Node: h1"
root@taoyucheng-dev:/home/taoyucheng# ipconfig
ipconfig: command not found
root@taoyucheng-dev:/home/taoyucheng# ifconfig

Command 'ifconfig' not found, did you mean:

  command 'ifconfig' from deb net-tools (1.60+git20180626.aebd88e-1ubuntu1)

Try: apt install <deb name>
root@taoyucheng-dev:/home/taoyucheng# ipconfig

Command 'ipconfig' not found, did you mean:

  command 'ifconfig' from deb net-tools (1.60+git20180626.aebd88e-1ubuntu1)
  command 'iwconfig' from deb wireless-tools (30~pre9-13ubuntu1)
  command 'iconfig' from deb ipmiutil (3.1.5-1)

Try: apt install <deb name>
root@taoyucheng-dev:/home/taoyucheng# ifconfig
h1-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.1 netmask 255.0.0.0 broadcast 10.255.255.255
    inet6 fe80::2415:c5ff:fe30:9589 prefixlen 64 scopeid 0x20<link>
    ether 26:15:c5:30:95:89 txqueuelen 1000 (Ethernet)
    RX packets 78 bytes 7300 (7.3 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 11 bytes 866 (866.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@taoyucheng-dev:/home/taoyucheng# iperf -s
-----
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[ 22] local 10.0.0.1 port 5001 connected with 10.0.0.2 port 42814
[ ID] Interval      Transfer      Bandwidth
[ 22] 0.0-10.0 sec   44.2 GBytes   38.0 Gbits/sec
```

Node:h2中 iperf的输出结果：

```
3月 21 10:54
"Node: h2"
root@taoyucheng-dev:/home/taoyucheng# iperf -c 10.0.0.1
-----
Client connecting to 10.0.0.1, TCP port 5001
TCP window size: 13.1 MByte (default)
-----
[ 21] local 10.0.0.2 port 42814 connected with 10.0.0.1 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 21] 0.0-10.0 sec  44.2 GBytes 38.0 Gbits/sec
root@taoyucheng-dev:/home/taoyucheng#
```

Task 2.1 网卡限速

Node:h1中 iperf的输出结果:

```
3月 21 11:15
"Node: h1"
root@taoyucheng-dev:/home/taoyucheng# iperf -u -s
-----
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 208 KByte (default)
-----
[ 21] local 10.0.0.1 port 5001 connected with 10.0.0.2 port 60898
[ ID] Interval      Transfer    Bandwidth      Jitter    Lost/Total Datagrams
[ 21] 0.0-10.3 sec  6.81 MBytes  5.57 Mbits/sec  15.695 ms 4063/ 8918 (46%)

```

Node:h2中 iperf的输出结果:

```
3月 21 11:15
"Node: h2"
root@taoyucheng-dev:/home/taoyucheng# iperf -u -c 10.0.0.1 -b 10M
-----
Client connecting to 10.0.0.1, UDP port 5001
Sending 1470 byte datagrams, IPG target: 1121.52 us (kalman adjust)
UDP buffer size: 208 KByte (default)
-----
[ 21] local 10.0.0.2 port 60898 connected with 10.0.0.1 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 21] 0.0-10.0 sec  12.5 MBytes  10.5 Mbits/sec
[ 21] Sent 8917 datagrams
[ 21] Server Report:
[ 21] 0.0-10.3 sec  6.81 MBytes  5.57 Mbits/sec  15.694 ms 4063/ 8918 (46%)
root@taoyucheng-dev:/home/taoyucheng#
```

Task 2.2 队列限速

Node:h3中 iperf的输出结果:

```
3月 21 11:32
"Node: h3"
root@taoyucheng-dev:/home/taoyucheng# iperf -u -c 10.0.0.4 -b 10M
-----
Client connecting to 10.0.0.4, UDP port 5001
Sending 1470 byte datagrams, IPG target: 1121.52 us (kalman adjust)
UDP buffer size: 208 KByte (default)
-----
[ 21] local 10.0.0.3 port 54608 connected with 10.0.0.4 port 5001
[ ID] Interval      Transfer      Bandwidth
[ 21] 0.0-10.1 sec  5.90 MBytes  4.92 Mbits/sec
[ 21] Sent 4211 datagrams
[ 21] Server Report:
[ 21] 0.0-10.2 sec  5.90 MBytes  4.86 Mbits/sec  58.175 ms   0/ 4211 (0%)
root@taoyucheng-dev:/home/taoyucheng#
```

Node:h4中 iperf的输出结果:

```
3月 21 11:32
"Node: h4"
root@taoyucheng-dev:/home/taoyucheng# ifconfig
h4-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.4 netmask 255.0.0.0 broadcast 10.255.255.255
    inet6 fe80::9c2e:37ff:feaf:ec1d prefixlen 64 scopeid 0x20<link>
    ether 9e:2e:37:af:ec:1d txqueuelen 1000 (Ethernet)
    RX packets 107 bytes 9494 (9.4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 15 bytes 1146 (1.1 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@taoyucheng-dev:/home/taoyucheng# iperf -u -s
-----
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 208 KByte (default)
-----
^[[A[ 21] local 10.0.0.4 port 5001 connected with 10.0.0.3 port 54608
[ ID] Interval      Transfer      Bandwidth      Jitter   Lost/Total Datagrams
[ 21] 0.0-10.2 sec  5.90 MBytes  4.86 Mbits/sec  58.176 ms   0/ 4211 (0%)
```

Task 2.3 Meter表限速

Node:h6中 iperf的输出结果:

3月 21 11:52

"Node: h6"

```
root@taoyucheng-dev:/home/taoyucheng# ifconfig
h6-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.6 netmask 255.0.0.0 broadcast 10.255.255.255
    inet6 fe80::c847:41ff:fed4:977a prefixlen 64 scopeid 0x20<link>
    ether ca:47:41:d4:97:7a txqueuelen 1000 (Ethernet)
    RX packets 4667 bytes 6898362 (6.8 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 28 bytes 16420 (16.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@taoyucheng-dev:/home/taoyucheng# iperf -u -s
-----
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 208 KByte (default)
-----
[ 21] local 10.0.0.6 port 5001 connected with 10.0.0.5 port 35179
[ ID] Interval      Transfer      Bandwidth      Jitter    Lost/Total Datagrams
[ 21]  0.0-10.0 sec   6.37 MBytes   5.35 Mbits/sec  0.041 ms  4371/ 8917 (49%)
```

Node:h5中 iperf的输出结果:

```
3月 21 11:52
"Node: h5"
root@taoyucheng-dev:/home/taoyucheng# ifconfig
h5-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.5 netmask 255.0.0.0 broadcast 10.255.255.255
    inet6 fe80::80a6:13ff:fe27:9b1e prefixlen 64 scopeid 0x20<link>
    ether 82:a6:13:27:9b:1e txqueuelen 1000 (Ethernet)
    RX packets 123 bytes 24950 (24.9 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 8945 bytes 13500366 (13.5 MB)
root@taoyucheng-dev:/home/taoyucheng# ifconfig
h5-eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.0.5 netmask 255.0.0.0 broadcast 10.255.255.255
    inet6 fe80::80a6:13ff:fe27:9b1e prefixlen 64 scopeid 0x20<link>
    ether 82:a6:13:27:9b:1e txqueuelen 1000 (Ethernet)
    RX packets 123 bytes 24950 (24.9 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 8945 bytes 13500366 (13.5 MB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@taoyucheng-dev:/home/taoyucheng# ethtool -K h5-eth0 tx off
Actual changes:
tx-checksumming: off
    tx-checksum-ip-generic: off
    tx-checksum-sctp: off
tcp-segmentation-offload: off
    tx-tcp-segmentation: off [requested on]
    tx-tcp-ecn-segmentation: off [requested on]
    tx-tcp-mangleid-segmentation: off [requested on]
    tx-tcp6-segmentation: off [requested on]
root@taoyucheng-dev:/home/taoyucheng# iperf -u -c 10.0.0.6 -b 10M
-----
Client connecting to 10.0.0.6, UDP port 5001
Sending 1470 byte datagrams, IPG target: 1121.52 us (kalman adjust)
UDP buffer size: 208 KByte (default)
-----
[ 21] local 10.0.0.5 port 35179 connected with 10.0.0.6 port 5001
[ 21] WARNING: did not receive ack of last datagram after 10 tries.
[ ID] Interval      Transfer      Bandwidth
[ 21]  0.0-10.0 sec  12.5 MBytes  10.5 Mbits/sec
[ 21] Sent 8917 datagrams
root@taoyucheng-dev:/home/taoyucheng#
```

Question 1 尝试理解Line19,20两条指令，指出每条指令的具体工作是什么，并逐个分析其中各个参数的具体含义。

- **line 19:** `ovs-ofctl add-flow s1 in_port=5,action=meter:1,output:6 -o openflow13`

用于对s1交换机添加下发流量表操作：数据包从5端口发入，经由meter表1处理后，再从6端口转出。其中各参数含义分析如下：

`add-flow`：是指添加下发流量表操作

`s1`：是指定的交换机

`in_port=5`：表示入端口为5

`action=meter:1,output:6`：指定转发动作，表示从出端口6转发出去之前，会先用meter表1进行处理。根据**line 11**可以看出，处理方式是丢弃掉超过5M的流量，然后再转发到6端口

`-o openflow13`：指定OpenFlow version为openflow13

- **line 20:** `ovs-ofctl dump-flows s1 -O openflow13`

用于查看s1交换机的流量表。其中各参数含义分析如下：

`dump-flows`：是指查看流量表操作

`s1`：是指定的交换机

`-O openflow13`：指定OpenFlow version为openflow13

Question 2 到这里，你已经完成了三种限速方式的实验，并获得了三组测试数据，请你就三组数据中的带宽、抖动和丢包率等参数，对三种限速方式进行横向比较，并适当地分析原因。

	bandwidth(Mbits/sec)	jitter(ms)	lost/total datagrams
网卡限速	5.57	15.694	46%
队列限速	4.86	58.176	0%
Meter表限速	5.35	0.041	49%

- 网卡限速会在接收分组的速率超过配置速率后，直接将超出带宽的数据包丢弃掉，因而会带来很高的丢包率。因而从本次实验的结果来看，其虽然能基本满足限速要求，但抖动较大，丢包率也很高。
- 队列限速设置有缓冲区，把超出带宽的数据包先缓存起来之后再进行处理，可以缓存和调度数据包发送顺序，比policing更加精确和有效。所以从本次实验的结果来看，其不仅对限速要求满足得最好，丢包率也相对最低（几乎不会丢包）。但也因为其会对数据包进行缓存起来平滑网络，以至于其抖动可能会很大。
- Meter表限速是SDN的限速方式，数据包基于其当前的速率会被Meter Band来处理，而超出带宽的包会被直接丢弃。因而从本次实验的结果来看，其虽然能基本满足限速要求，抖动也不大，但丢包率也不低。

Task3 在限制Server端（h1）的带宽为10Mb的前提下，观察稳定后的三个Client的带宽，将结果截图并简单分析。

```
#使用队列queue将server的带宽限制到10Mb
ovs-vsctl set port s1-eth1 qos=@newqos -- \
--id=@newqos create qos type=linux-htb queues=0=@q0 -- \
--id=@q0 create queue other-config:max-rate=10000000
```

Node:h1中观察得到三个client的输出结果分别为：

```
3月 21 23:37
"Node: h1"
root@taoyucheng-dev:/home/taoyucheng# iperf -u -s
-----
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 208 KByte (default)
-----
[ 21] local 10.0.0.1 port 5001 connected with 10.0.0.2 port 48103
[ 22] local 10.0.0.1 port 5001 connected with 10.0.0.3 port 33941
[ 23] local 10.0.0.1 port 5001 connected with 10.0.0.4 port 33256
[ ID] Interval      Transfer    Bandwidth    Jitter    Lost/Total Datagrams
[ 21] 0.0-20.2 sec  8.07 MBytes  3.35 Mbits/sec  19.803 ms    0/ 5759 (0%)
[ 22] 0.0-20.2 sec  7.88 MBytes  3.28 Mbits/sec   9.468 ms    0/ 5619 (0%)
[ 23] 0.0-19.9 sec  7.92 MBytes  3.35 Mbits/sec   4.865 ms    0/ 5653 (0%)
[ 23] 0.0000-19.8527 sec  1 datagrams received out-of-order
```

可见3个client的带宽大小几乎相同，均为3.3Mb左右。这是因为三个client的结点均未做任何配置和限速，在同样的条件下同时运行，因而三者竞争力几乎相同，平分server的10Mb带宽。

Task4 你可以通过上述三种限速的方法来达成目标，请记录你的设计过程（思路及运行指令），并将你稳定后的三个Client的带宽结果截图。

结合前面的实验内容，我决定采用Meter表限速的方式对3个Client的带宽进行限制。

考虑到要求h2在5Mb及以上，h3在3Mb及以上，h4在保证h2和h3的前提下尽量多。因而我决定将h2和h3的带宽分别限制在不超过5.1Mb和3.1Mb，而h4的带宽也就限制在 $10-5.1-3.1=1.8$ Mb的大小，以保证h2和h3都能始终包有不低于5Mb和3Mb的带宽。

```
ovs-ofctl del-meters s1 -O OpenFlow13 #删除之前配置的meter表
ovs-ofctl -O OpenFlow13 dump-meters s1 #查看meter表
#配置meter表
ovs-ofctl add-meter s1 meter=1,kbps,band=type=drop,rate=5100 -O OpenFlow13
ovs-ofctl add-meter s1 meter=2,kbps,band=type=drop,rate=3100 -O OpenFlow13
ovs-ofctl add-meter s1 meter=3,kbps,band=type=drop,rate=1800 -O OpenFlow13
ovs-ofctl add-flow s1 in_port=2,action=meter:1,output:1 -O openflow13
ovs-ofctl add-flow s1 in_port=3,action=meter:2,output:1 -O openflow13
ovs-ofctl add-flow s1 in_port=4,action=meter:3,output:1 -O openflow13
ovs-ofctl -O OpenFlow13 dump-meters s1 #查看meter表

ethtool -K h2-eth0 tx off      #在h2中关闭网卡的tx校验
ethtool -K h3-eth0 tx off      #在h3中关闭网卡的tx校验
ethtool -K h4-eth0 tx off      #在h4中关闭网卡的tx校验
```

Node:观察得到的输出结果：

```
3月 25 11:32

"Node: h3"
[ 21] 0.0- 1.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 1.0- 2.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 2.0- 3.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 3.0- 4.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 4.0- 5.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 5.0- 6.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 6.0- 7.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 7.0- 8.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 8.0- 9.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 9.0-10.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 10.0-11.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 11.0-12.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 12.0-13.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 13.0-14.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 14.0-15.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 15.0-16.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 16.0-17.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 17.0-18.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 18.0-19.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 0.0-20.0 sec 25.0 MBytes 10.5 Mbits/sec
[ 21] Sent 17833 datagrams
[ 21] Server Report:
[ 21] 0.0-20.0 sec 7.54 MBytes 3.12 Mbits/sec 15.706 ms 12453/17834 (70%)
root@taoyucheng-dev:/home/taoyucheng#

"Node: h1"
root@taoyucheng-dev:/home/taoyucheng# iperf -u -s
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 208 KByte (default)
[ 21] local 10.0.0.1 port 5001 connected with 10.0.0.2 port 49592
[ 22] local 10.0.0.1 port 5001 connected with 10.0.0.3 port 55678
[ 23] local 10.0.0.1 port 5001 connected with 10.0.0.4 port 45828
[ 10] Interval      Transfer      Bandwidth      Jitter    Lost/Total Datagrams
[ 21] 0.0-20.3 sec 12.4 MBytes  5.14 Mbits/sec 15.682 ms 8981/17835 (50%)
[ 22] 0.0-20.3 sec 7.54 MBytes  3.12 Mbits/sec 15.707 ms 12453/17834 (70%)
[ 23] 0.0-20.0 sec 4.38 MBytes  1.84 Mbits/sec  0.022 ms 14710/17834 (82%)

"Node: h4"
[ 21] 0.0- 1.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 1.0- 2.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 2.0- 3.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 3.0- 4.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 4.0- 5.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 5.0- 6.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 6.0- 7.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 7.0- 8.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 8.0- 9.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 9.0-10.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 10.0-11.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 11.0-12.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 12.0-13.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 13.0-14.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 14.0-15.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 15.0-16.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 16.0-17.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 17.0-18.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 18.0-19.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 0.0-20.0 sec 25.0 MBytes 10.5 Mbits/sec
[ 21] Sent 17834 datagrams
[ 21] Server Report:
[ 21] 0.0-20.0 sec 4.38 MBytes  1.84 Mbits/sec  0.022 ms 14710/17834 (82%)
root@taoyucheng-dev:/home/taoyucheng#

"Node: h2"
[ 21] 0.0- 1.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 1.0- 2.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 2.0- 3.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 3.0- 4.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 4.0- 5.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 5.0- 6.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 6.0- 7.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 7.0- 8.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 8.0- 9.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 9.0-10.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 10.0-11.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 11.0-12.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 12.0-13.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 13.0-14.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 14.0-15.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 15.0-16.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 16.0-17.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 17.0-18.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 18.0-19.0 sec 1.25 MBytes 10.5 Mbits/sec
[ 21] 0.0-20.0 sec 25.0 MBytes 10.5 Mbits/sec
[ 21] Sent 17834 datagrams
[ 21] Server Report:
[ 21] 0.0-20.3 sec 12.4 MBytes  5.14 Mbits/sec 15.682 ms 8981/17835 (50%)
root@taoyucheng-dev:/home/taoyucheng#
```

可以看出，我们成功满足了h2在5Mb及以上，h3在3Mb及以上的要求。