

eStadium Project

Lab 2: Iperf Command

Objectives

- Being familiar with the command *iperf*.

In this Lab, we will set up two computers (PC1 and PC2) as an ad-hoc network and use the command *iperf* to measure network parameters of TCP (Transmission Control Protocol) connection and UDP (User Datagram Protocol) connection.

Mainly, we use *iperf* on TCP connection to measure the TCP throughput (Part B), which will lead to calculate the TCP window size. On the other hand, we use *iperf* on UDP connection to measure UDP throughput, jitter, and data loss (Part C). In addition, there are some *iperf*'s options that we will take a look (Part D – G).

Remark:

- One PC will act as a server (=receiver).
- Another PC will act as a client (=transmitter).
- The recorded measurement values are the values shown on the server, not the client.
- The setup with more than two PCs is possible.
- To install *iperf*, on Ubuntu, type

```
>> sudo apt-get install iperf
```

Note: This will install *iperf* online, then make sure that the internet must be connected during the installation.

- Please see the references for comprehensive explanation and examples.
- Please see the references or type *man iperf* to see the *iperf* command's options.
- If not specified, just put two PCs apart with a convenient distance (1-2m).

A: Setup the Ad-hoc Network

A.1) Set up the ad-hoc network between two PCs by following the steps in Lab 1. Note that let the IP address of PC1 is *PC1_IPaddress* (for example, 192.168.12.1).

B: TCP Connection – TCP Throughput Measurement

B.1) Set PC1 as a server (receiver) by typing

```
>> iperf -s
```

B.2) Set PC2 as a client (transmitter) by typing

```
>> iperf -c PC1_IPaddress
```

The screens are shown below

The image shows two terminal windows. The top window, titled 'seksan@ubuntu: ~', shows the output of the command 'iperf -s'. It indicates the server is listening on TCP port 5001 with a window size of 85.3 KByte. A connection is established from 192.168.12.12 port 50608, showing a transfer of 6.08 MBytes and a bandwidth of 4.92 Mbits/sec over a 10-second interval. The bottom window, titled 'estadium@ubuntu: ~', shows the output of 'iperf -c 192.168.12.1'. It shows the client connecting to 192.168.12.1 on TCP port 5001 with a window size of 16.0 KByte. The connection to 192.168.12.12 port 50608 shows a transfer of 6.08 MBytes and a bandwidth of 5.01 Mbits/sec over a 10-second interval.

```
seksan@ubuntu:~$ iperf -s
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)

[ 4] local 192.168.12.1 port 5001 connected with 192.168.12.12 port 50608
[ ID] Interval      Transfer      Bandwidth
[ 4]  0.0-10.4 sec  6.08 MBytes  4.92 Mbits/sec

estadium@ubuntu:~$ iperf -c 192.168.12.1
Client connecting to 192.168.12.1, TCP port 5001
TCP window size: 16.0 KByte (default)

[ 3] local 192.168.12.12 port 50608 connected with 192.168.12.1 port 5001
[ ID] Interval      Transfer      Bandwidth
[ 3]  0.0-10.2 sec  6.08 MBytes  5.01 Mbits/sec
```

Remark:

- The server must be set up first.
- The measured TCP throughput is 4.92 Mbit/s (the throughput shown on the server's screen).
- By default, *iperf* runs a 10 second test.
- By default, TCP window is 85.3 KByte at the server.
- By default, TCP window is 16.0 KByte at the client.
- When we know the TCP throughput, we can calculate an appropriate TCP window size (of the client) following this equation:

$$\text{TCP window size} = (\text{Throughput}) \times (\text{Round Trip Time})$$
 where the round trip time (RTT) is measured by using *ping* command.

C: UDP Connection – UDP Throughput, Jitter, and Loss Measurement

C.1) Set PC1 as a server (receiver) by typing

```
>> iperf -s -u
```

C.2) Set PC2 as a client (transmitter) by typing

```
>> iperf -c PC1_IPaddress -u
```

The client will send the UDP packets with bandwidth 1 Mb/s (default). The screens are shown below

The image shows a terminal window titled 'seksan@ubuntu: ~' with the output of 'iperf -s -u'. It indicates the server is listening on UDP port 5001 with a buffer size of 110 KByte. A connection is established from 192.168.12.12 port 60475, showing a transfer of 1.25 MBytes and a bandwidth of 1.05 Mbits/sec, with a jitter of 1.033 ms and 0% loss over a 10-second interval.

```
seksan@ubuntu:~$ iperf -s -u
Server listening on UDP port 5001
Receiving 1470 byte datagrams
UDP buffer size: 110 KByte (default)

[ 3] local 192.168.12.1 port 5001 connected with 192.168.12.12 port 60475
[ ID] Interval      Transfer      Bandwidth      Jitter    Lost/Total Datagrams
[ 3]  0.0-10.0 sec  1.25 MBytes  1.05 Mbits/sec  1.033 ms   0/ 893 (0%)
```

```
estadium@ubuntu: ~  
File Edit View Terminal Tabs Help  
estadium@ubuntu:~$ iperf -c 192.168.12.1 -u  
-----  
Client connecting to 192.168.12.1, UDP port 5001  
Sending 1470 byte datagrams  
UDP buffer size: 110 KByte (default)  
-----  
[ 3] local 192.168.12.12 port 60475 connected with 192.168.12.1 port 5001  
[ ID] Interval      Transfer      Bandwidth  
[ 3] 0.0-10.0 sec  1.25 MBytes  1.05 Mbits/sec  
[ 3] Sent 893 datagrams  
[ 3] Server Report:  
[ ID] Interval      Transfer      Bandwidth      Jitter    Lost/Total Datagrams  
[ 3] 0.0-10.0 sec  1.25 MBytes  1.05 Mbits/sec  1.033 ms   0/ 893 (0%)  
estadium@ubuntu:~$
```

Remark:

- The server must be set up first.
- The measured UDP throughput is the throughput shown on the server's screen.
By default the UDP throughput is around 1 Mbit/s.
- The measured UDP jitter is shown on the server's screen, which is 1.033 ms.
- The measured data loss is shown on the server's screen, which is 0.
- By default, *iperf* runs a 10 second test.
- By default, UDP buffer is 110 KByte at the server.
- By default, UDP buffer is 110 KByte at the client.

D: Iperf's Option - TCP Connection with a Varying Observed Time Duration

D.1) Set PC1 as a server (receiver) by typing

```
>> iperf -s
```

D.2) Set PC2 as a client (transmitter) by typing

```
>> iperf -c PC1_IPaddress -t 5
```

The client will send TCP packets for 5 second. Note that this option is also available for UDP connection. The screens are shown below.

```
seksan@ubuntu: ~  
File Edit View Terminal Tabs Help  
seksan@ubuntu:~$ iperf -s  
-----  
Server listening on TCP port 5001  
TCP window size: 85.3 KByte (default)  
-----  
[ 4] local 192.168.12.1 port 5001 connected with 192.168.12.12 port 57372  
[ ID] Interval      Transfer      Bandwidth  
[ 4] 0.0- 5.2 sec  5.05 MBytes  8.22 Mbits/sec
```

```
estadium@ubuntu: ~  
File Edit View Terminal Tabs Help  
estadium@ubuntu:~$ iperf -c 192.168.12.1 -t 5  
-----  
Client connecting to 192.168.12.1, TCP port 5001  
TCP window size: 10.0 KByte (default)  
-----  
[ 3] local 192.168.12.12 port 57372 connected with 192.168.12.1 port 5001  
[ ID] Interval      Transfer      Bandwidth  
[ 3] 0.0- 5.1 sec  5.05 MBytes  8.38 Mbits/sec  
estadium@ubuntu:~$
```

E: Iperf's Option - TCP Connection with a Varying Observed Time Interval

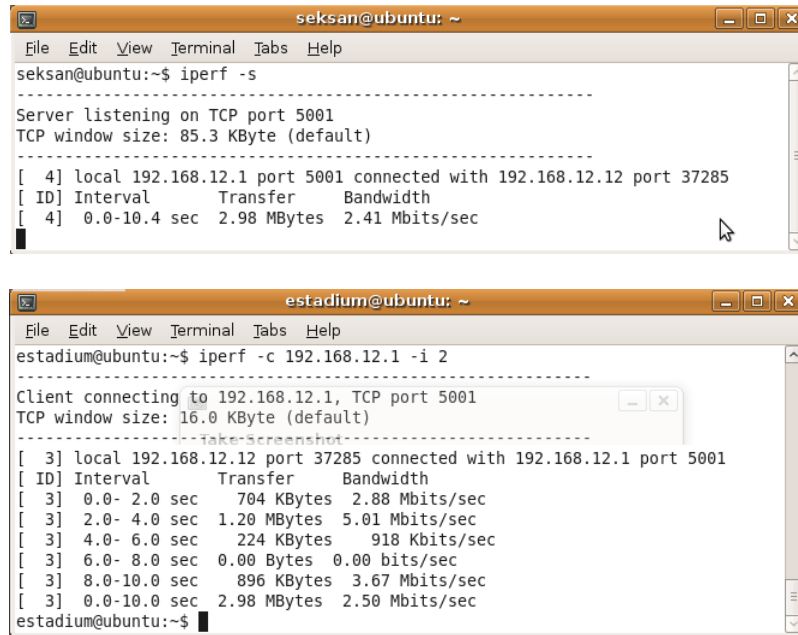
E.1) Set PC1 as a server (receiver) by typing

```
>> iperf -s
```

E.2) Set PC2 as a client (transmitter) by typing

```
>> iperf -c PC1_IPaddress -i 2
```

The client will send TCP packets for 10 second (default) and show the details every 2 second. Note that this option is also available for UDP connection. The screens are shown below.



The image contains two terminal window screenshots. The top window, titled 'seksan@ubuntu: ~', shows the server output for 'iperf -s'. It indicates listening on TCP port 5001 and shows a connection from 192.168.12.12 port 37285 with a bandwidth of 2.41 Mbits/sec. The bottom window, titled 'estadium@ubuntu: ~', shows the client output for 'iperf -c 192.168.12.1 -i 2'. It shows the client connecting to 192.168.12.1 port 5001 and displays a table of performance metrics over 10 seconds.

```
seksan@ubuntu: ~
File Edit View Terminal Tabs Help
seksan@ubuntu:~$ iperf -s
-----
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[ 4] local 192.168.12.1 port 5001 connected with 192.168.12.12 port 37285
[ ID] Interval      Transfer    Bandwidth
[ 4] 0.0-10.4 sec  2.98 MBytes  2.41 Mbits/sec

estadium@ubuntu: ~
File Edit View Terminal Tabs Help
estadium@ubuntu:~$ iperf -c 192.168.12.1 -i 2
-----
Client connecting to 192.168.12.1, TCP port 5001
TCP window size: 16.0 KByte (default)
-----
[ 3] local 192.168.12.12 port 37285 connected with 192.168.12.1 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 3] 0.0- 2.0 sec   704 KBytes  2.88 Mbits/sec
[ 3] 2.0- 4.0 sec   1.20 MBytes  5.01 Mbits/sec
[ 3] 4.0- 6.0 sec    224 KBytes   918 Kbits/sec
[ 3] 6.0- 8.0 sec    0.00 Bytes   0.00 bits/sec
[ 3] 8.0-10.0 sec    896 KBytes  3.67 Mbits/sec
[ 3] 0.0-10.0 sec   2.98 MBytes  2.50 Mbits/sec
estadium@ubuntu:~$
```

F: Iperf's Option - TCP Connection with a Varying TCP Window Size

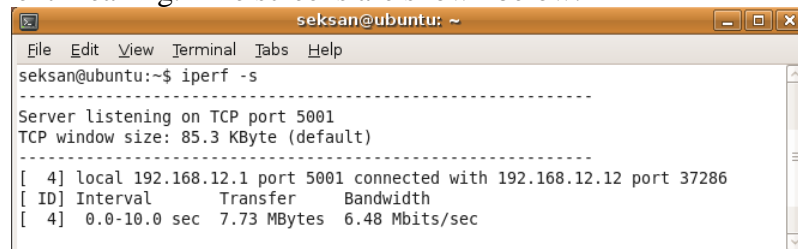
F.1) Set PC1 as a server (receiver) by typing

```
>> iperf -s
```

F.2) Set PC2 as a client (transmitter) by typing

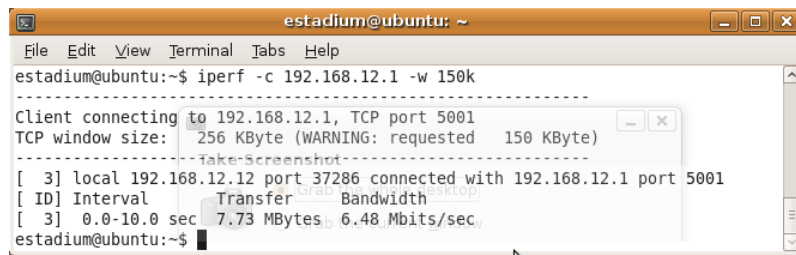
```
>> iperf -c PC1_IPaddress -w 150k
```

The client will send TCP packets for 10 second (default) with requesting the TCP window size 150 KByte. Note that this option is also available for UDP connection but with a different meaning. The screens are shown below.



The image shows a terminal window titled 'seksan@ubuntu: ~' running 'iperf -s'. It shows the server listening on TCP port 5001 and receiving a connection from 192.168.12.12 port 37286. The output shows a bandwidth of 6.48 Mbits/sec over a 10-second interval.

```
seksan@ubuntu: ~
File Edit View Terminal Tabs Help
seksan@ubuntu:~$ iperf -s
-----
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[ 4] local 192.168.12.1 port 5001 connected with 192.168.12.12 port 37286
[ ID] Interval      Transfer    Bandwidth
[ 4] 0.0-10.0 sec  7.73 MBytes  6.48 Mbits/sec
```



```
estadium@ubuntu: ~  
File Edit View Terminal Tabs Help  
estadium@ubuntu:~$ iperf -c 192.168.12.1 -w 150k  
-----  
Client connecting to 192.168.12.1, TCP port 5001  
TCP window size: 256 KByte (WARNING: requested 150 KByte)  
-----  
[ 3] local 192.168.12.12 port 37286 connected with 192.168.12.1 port 5001  
[ ID] Interval      Transfer    Bandwidth  
[ 3] 0.0-10.0 sec  7.73 MBytes 6.48 Mbits/sec  
estadium@ubuntu:~$
```

G: Iperf's Option - UDP Connection with a Varying Bandwidth

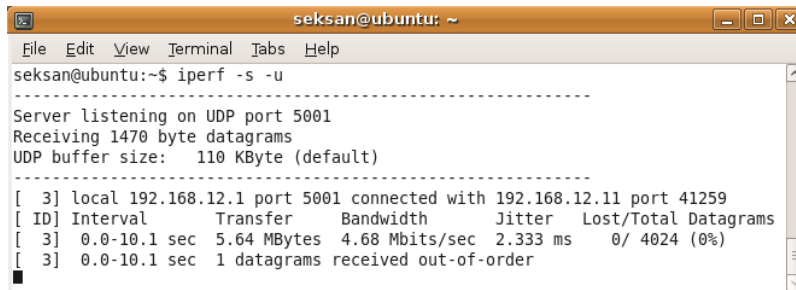
G.1) Set PC1 as a server (receiver) by typing

```
>> iperf -s -u
```

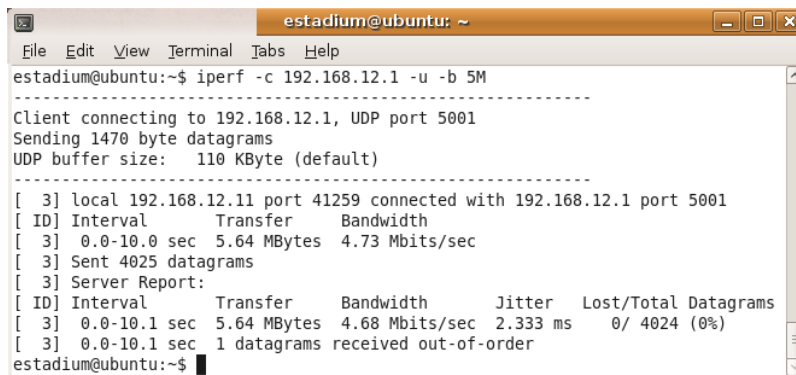
G.2) Set PC2 as a client (transmitter) by typing

```
>> iperf -c PC1_IPaddress -u -b 5M
```

The client will send UDP packets for 10 second (default) with the bandwidth 5 Mbit/s.
The screens are shown below.



```
seksan@ubuntu: ~  
File Edit View Terminal Tabs Help  
seksan@ubuntu:~$ iperf -s -u  
-----  
Server listening on UDP port 5001  
Receiving 1470 byte datagrams  
UDP buffer size: 110 KByte (default)  
-----  
[ 3] local 192.168.12.1 port 5001 connected with 192.168.12.11 port 41259  
[ ID] Interval      Transfer    Bandwidth    Jitter  Lost/Total Datagrams  
[ 3] 0.0-10.1 sec  5.64 MBytes 4.68 Mbits/sec 2.333 ms  0/ 4024 (0%)  
[ 3] 0.0-10.1 sec  1 datagrams received out-of-order  
■
```



```
estadium@ubuntu: ~  
File Edit View Terminal Tabs Help  
estadium@ubuntu:~$ iperf -c 192.168.12.1 -u -b 5M  
-----  
Client connecting to 192.168.12.1, UDP port 5001  
Sending 1470 byte datagrams  
UDP buffer size: 110 KByte (default)  
-----  
[ 3] local 192.168.12.11 port 41259 connected with 192.168.12.1 port 5001  
[ ID] Interval      Transfer    Bandwidth  
[ 3] 0.0-10.0 sec  5.64 MBytes 4.73 Mbits/sec  
[ 3] Sent 4025 datagrams  
[ 3] Server Report:  
[ ID] Interval      Transfer    Bandwidth    Jitter  Lost/Total Datagrams  
[ 3] 0.0-10.1 sec  5.64 MBytes 4.68 Mbits/sec 2.333 ms  0/ 4024 (0%)  
[ 3] 0.0-10.1 sec  1 datagrams received out-of-order  
estadium@ubuntu:~$
```

References:

- [1] <http://openmaniak.com/iperf.php>
- [2] <http://pirlwww.lpl.arizona.edu/resources/guide/software/iperf/>
- [3] <http://kb.pert.geant2.net/PERTKB/IperfTool>