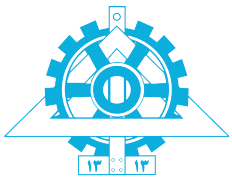
TCP and Its Applications[1]

HARDNESS : 8/10



University of Tehran
School of Electrical and Computer Engineering

دانشگاه تهران
دانشکده‌ی مهندسی برق و کامپیوتر

Computer Network Lab

آزمایشگاه شبکه‌های کامپیوتری

Professor:
Dr. Ahmad Khonsari
دکتر احمد خونساری
a_khonsari@ut.ac.ir

Amir Haji Ali Khamseh'i
امیر حاجی‌علی‌خمسه‌ء
khamse@ut.ac.ir

Reza Sharifnia
رضا شریف نیا
Reza.sharifnia@ut.ac.ir

Muhammad Borhani
محمد برهانی
borhani.m@ut.ac.ir

AmirAhmad Khordadi
امیراحمد خردادی
a.a.khordadi@ut.ac.ir

Sina Kashipazha
سینا کاشی‌پزها
sina_kashipazha@ut.ac.ir

Hadi Safari
هادی صفری
hadi.safari@ut.ac.ir

October 24, 2021

۲ آبان ۱۴۰۰

[1]S. Panwar, S. Mao, J.-dong Ryoo, and Y. Li, "TCP study," in TCP/IP Essentials: A Lab-Based Approach, Cambridge: Cambridge University Press, 2004, pp. 111–133.

Objectives

- TCP connection establishment and termination.
- TCP timers.
- TCP timeout and retransmission.
- TCP interactive data flow, using **telnet** as an example.
- TCP bulk data flow, using **socket** as a traffic generator.
- Further comparison of TCP and UDP.
- Tuning the TCP/IP kernel.

Part I

Exercises on TCP Connection Control

Like previous lab, connect two host with one hub together (Figure 5.0) or use Figure 1.3 with two host.

1 Telnet terminal

While `tcpdump -S h0.netlab and h1.netlab` is running, execute: `telnet h1.netlab echo` on the *h0* host, then type some text. Save the `tcpdump` output.

h_0's Console

```
tcpdump host 128.238.61.100 and 128.238.61.101  # or run wireshark
```

h_0's Auxiliary Console

```
telnet 128.238.61.101 echo -e q
```

Note: We set the *q* character as an *escape* character. So, do not type q character.

At the end, you can close the telnet connection by typing *q* character and run *close* command.

h_0's Auxiliary Console

```
q
close
```

Report

1. Explain TCP connection establishment and termination using the `tcpdump` output.
2. What were the announced MSS values for the two hosts?
3. What happens if there is an intermediate network that has an MTU less than the MSS of each host? See if the `DF`[1] flag was set in `tcpdump` output.

   You can change interface `MTU` with below command:

   h_0's Console

   ```
   sudo ifconfig eth0 mtu 68
   ```

[1] Don't Fragment

h_1's Console

```
sudo ifconfig eth0 mtu 68
```

2 TCP vs UDP Connection Establishment

While `tcpdump -nx host h0.netlab and h1.netlab` is running, use `socket`[2] to send a UDP datagram to the *h1* host from the *h0* machine:

h_0's Console

```
tcpdump -nx host 128.238.61.100 and 128.238.61.101  # or run wireshark
```

h_0's Auxiliary Console

```
socket -u -i -n1 128.238.61.101 8888
```

Save the `tcpdump` or `wireshark` output for your lab report.
Restart the above `tcpdump` command, execute `socket` in the TCP mode:

h_0's Console

```
tcpdump -nx host 128.238.61.100 and 128.238.61.101  # or run wireshark
```

h_0's Auxiliary Console

```
socket -i -n1 128.238.61.101 8888
```

Save the `tcpdump` output for your lab report.

Report

1. Explain what happened in both the UDP and TCP cases. When a client requests a non-existing server, how do UDP and TCP handle this request, respectively?

Part II

Exercise on TCP Interactive Data Flow

3 Interactive Data Flow

While `tcpdump` or Wireshark capture the traffic between your machine and a remote machine, issue the following commands:

h_0's Console

```
tcpdump -nv # or run wireshark
```

h_0's Auxiliary Console

```
telnet 128.238.61.101
```

[2] Basic command is `sock`. Use alternative `socket` (linked to `sock`).

Enter "netlab" as username and password for login in the remote host. After logging in to the host, type date and press the Enter key.

Now, in order to generate data faster than the round-trip time of a single byte to be sent and echoed, type any sequence of keys in the telnet window very rapidly.[3]

Save the tcpdump or Wireshark output for your lab report.

Report

Answer the following questions, based upon the tcpdump or Wireshark output saved in the above exercise.

1. What is a delayed acknowledgement? What is it used for?
2. Can you see any delayed acknowledgements in your tcpdump or Wireshark output?

   If yes, explain the reason. Mark some of the lines with delayed acknowledgements, and submit the tcpdump or Wireshark output with your report.

   Explain how the delayed ACK timer operates from your tcpdump or Wireshark output.

   If you don't see any delayed acknowledgements, explain the reason why none was observed.
3. What is the *Nagle*[4] algorithm used for?

   From your tcpdump or Wireshark output, can you tell whether the *Nagle* algorithm is enabled or not? Give the reason for your answer.

   From your tcpdump or Wireshark output for when you typed very rapidly, can you see any segment that contains more than one character going from your host to the remote machine?

   Now add link delay in the simulator [5] and do the same experiment again. Can you tell whether the Nagle algorithm is enabled or not? When you typed very rapidly, can you see any segment that contains more than one character going from your host to the remote machine?

Part III

Exercise on TCP Bulk Data Flow

4 IP Segment

Run tcpdump on your host by:

h_0's Console

```
tcpdump host 128.238.61.100 and 128.238.61.101  # or run wireshark
```

While tcpdump is running and capturing the packets between your machine and a remote machine, on the remote machine, which acts as the server, execute:

h_1's Console

```
socket -i -s 7777
```

Then, on your machine's which acts as the client, execute:

h_0's Auxiliary Console

```
socket -i -n16 128.238.61.101 7777
```

[3] For example hold "A" key or write "qwertyuiop" in telnet window.
[4] Nagle Algorithm is a means of improving the efficiency of TCP/IP networks by reducing the number of packets that need to be sent over the network.
[5] In GNS3, right click on link, select Packet Filter and set link delay to *100 ms*

Do the same experiment three times.

Save all the `tcpdump` outputs for your lab report.

Report

1. Using one of three `tcpdump` outputs, explain the operation of TCP in terms of data segments and their acknowledgements. Does the number of data segments differ from that of their acknowledgements?

   Compare all the `tcpdump` outputs you saved. Discuss any differences among them, in terms of data segments and their acknowledgements.

2. From the `tcpdump` output, how many different TCP flags can you see? Enumerate the flags and explain their meanings.

   How many different TCP options can you see? Explain their meanings.

Part IV

Exercises on TCP Timers and Retransmission

5 Keepalive parameter

Execute `sysctl -A | grep keepalive` to display the default values of the TCP kernel parameters that are related to the TCP keepalive timer.

h_0's Console

```
sysctl -A | grep keepalive
```

Report

1. What is the default value of the TCP keepalive timer?
2. What is the maximum number of TCP keepalive probes a host can send?

6 TCP Retransmission

Run `tcpdump` on your host by:

h_0's Console

```
tcpdump host 128.238.61.100 and 128.238.61.101     # or run Wireshark
```

While `tcpdump` or Wireshark is running to capture the packets between your host and a remote host, start a `socket` server on the remote host,

h_1's Console

```
socket -s 8888
```

Then add link delay in the simulator[6] and execute the below command on your host's Auxiliary Console,

h_0's Auxiliary Console

```
socket -i -n200 -p 100 128.238.61.101 8888
```

[6] In GNS3, `right click` on link, select `Packet Filter` and set link `delay` to *100 ms*

While the sender is injecting data segments into the network, shutdown the network interface on the *remote-host* that connect the sender to the hub for about ten seconds.

h_1's Auxiliary Console

```
# use one of below commands
ip link set eth0 down
ifconfig eth0 down
```

After observing several retransmissions, set network interface up: After seconds...

h_1's Auxiliary Console

```
# use one of below commands
ip link set eth0 up
ifconfig eth0 up
```

When all the data segments are sent, save the `tcpdump` or Wireshark output for the lab report.

Report

1. Submit the `tcpdump` or Wireshark output saved in this exercise.
2. From the `tcpdump` or Wireshark output, identify when the cable was disconnected.
3. Describe how the retransmission timer changes after sending each retransmitted packet, during the period when the cable was disconnected.
4. Explain how the number of data segments that the sender transmits at once (before getting an ACK) changes after the connection is reestablished[7].

Part V
Other Exercises

7 Fragmentation

Run `tcpdump` on your host by:

h_0's Console

```
tcpdump src host 128.238.61.100 # or run Wireshark
```

While `tcpdump` or Wireshark is running, execute the following command, which is similar to the command we used to find out the maximum size of a UDP datagram in the previous lab session (Chapter 5 of reference book),

h_0's Auxiliary Console

```
socket -i -n1 -w 70080 128.238.61.101 echo
```

Note: *70080* is larger than the maximum UDP datagram size we found in previous lab session.

Report

1. Did you observe any IP fragmentation?
2. If IP fragmentation did not occur this time, how do you explain this compared to what you observed in previous lab session for UDP packets?

[7] Can see TCP **window scale** option and **RTT**

8 Linux TCP/IP Kernel Parameter

Study the manual page of `/sbin/sysctl`.

h_0's Console

```
man /sbin/sysctl
```

Examine the default values of some TCP/IP configuration parameters that you might be interested in.

Examine the configuration files in the `/proc/sys/net/ipv4` directory.

h_0's Console

```
cd /proc/sys/net/ipv4
```

Note: You can see the directory's files and folders by:

h_0's Console

```
ls
```

Also, you can see files contents with the below command:

h_0's Console

```
cat file-name
```

Report

1. Explain what is `sysctl` command for?
2. Explain two arbitrary TCP/IP configuration parameters. What is their default values?
3. Name two arbitrary file in the `/proc/sys/net/ipv4` directory. What is their content?