



**Worksheet Jaringan Komunikasi Data
(CSIM603154)– 2020-2021 Gasal**

Week : 3

Topic : Applciation Layer: HTTP persistent with
Telnet

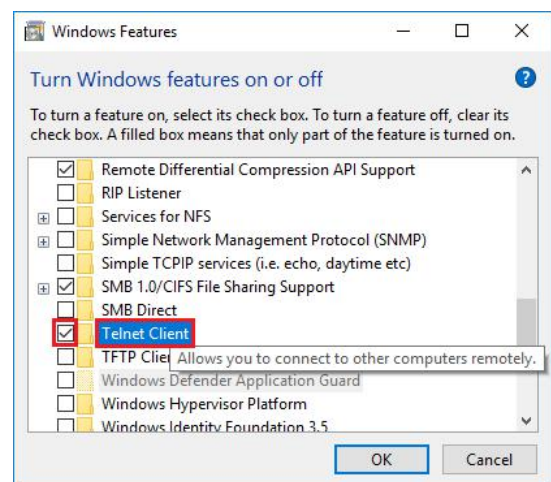
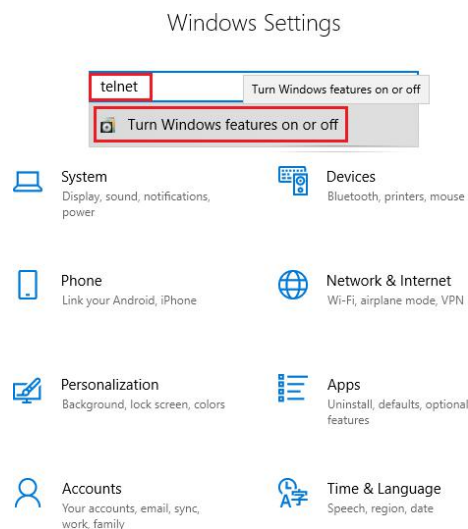
Lecturer : 1. Ari Wibisono
2. Muhammad Anwar Ma'sum

Name :
NPM :

A. Pre-requisite

You have to have a Telnet program in your
computer. NOTE:

- For Linux users, you can usually find Telnet already installed in your system.
Otherwise, you need to install it (the command varies depending on the Linux distribution).
 - o NOTE: This guide uses telnet in Linux Lubuntu.
- For Windows users, you need to enable telnet client. Go to Windows Settings > search for telnet
> go to "Turn Windows features on or off" > tick the "Telnet Client" to enable it.
 - o NOTE: in my Windows, I can't see the characters as I typed the command inside the telnet (after the telnet connection is executed).



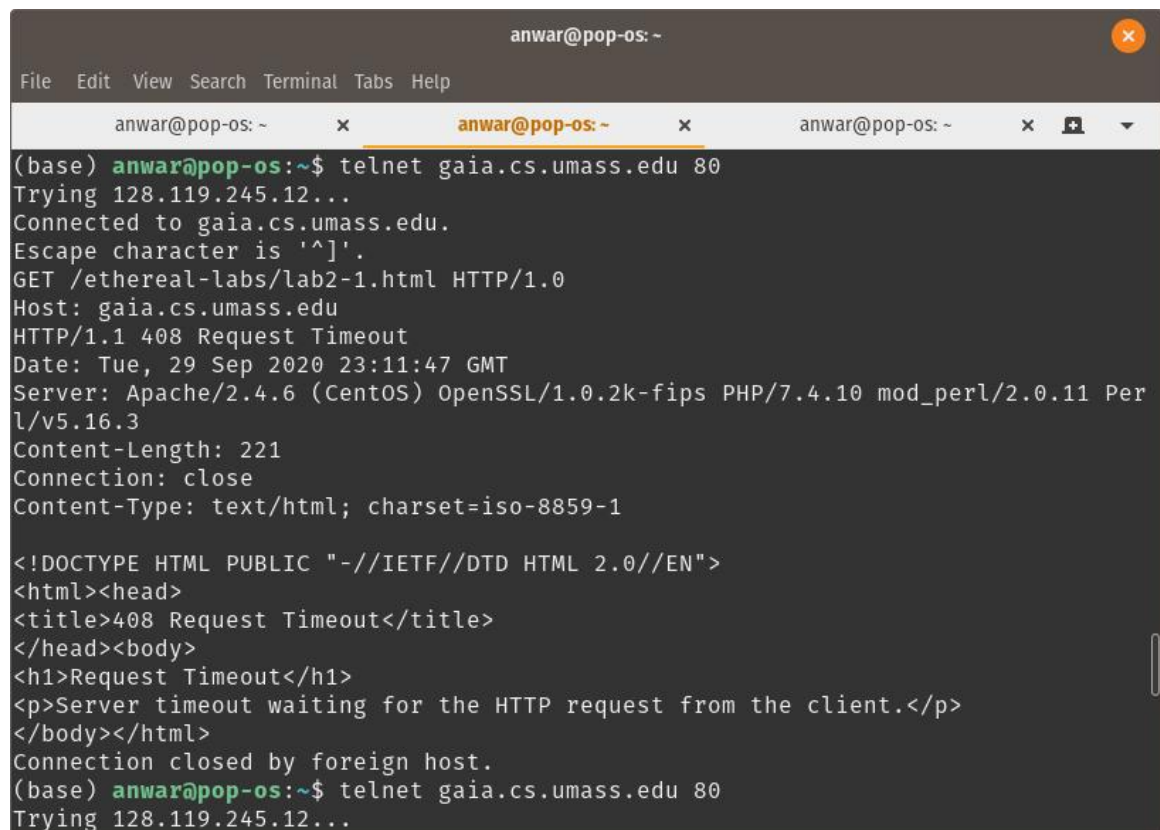
- For MacOS users, Telnet is still part of the system for the older version of MacOS.
 - o But it has been removed in the modern versions, e.g. Mojave & High Sierra. So you need to install it first (search for the instructions in the internet as I am not a Mac user).

B. Run Wireshark

1. Start wireshark and then start packet capture.

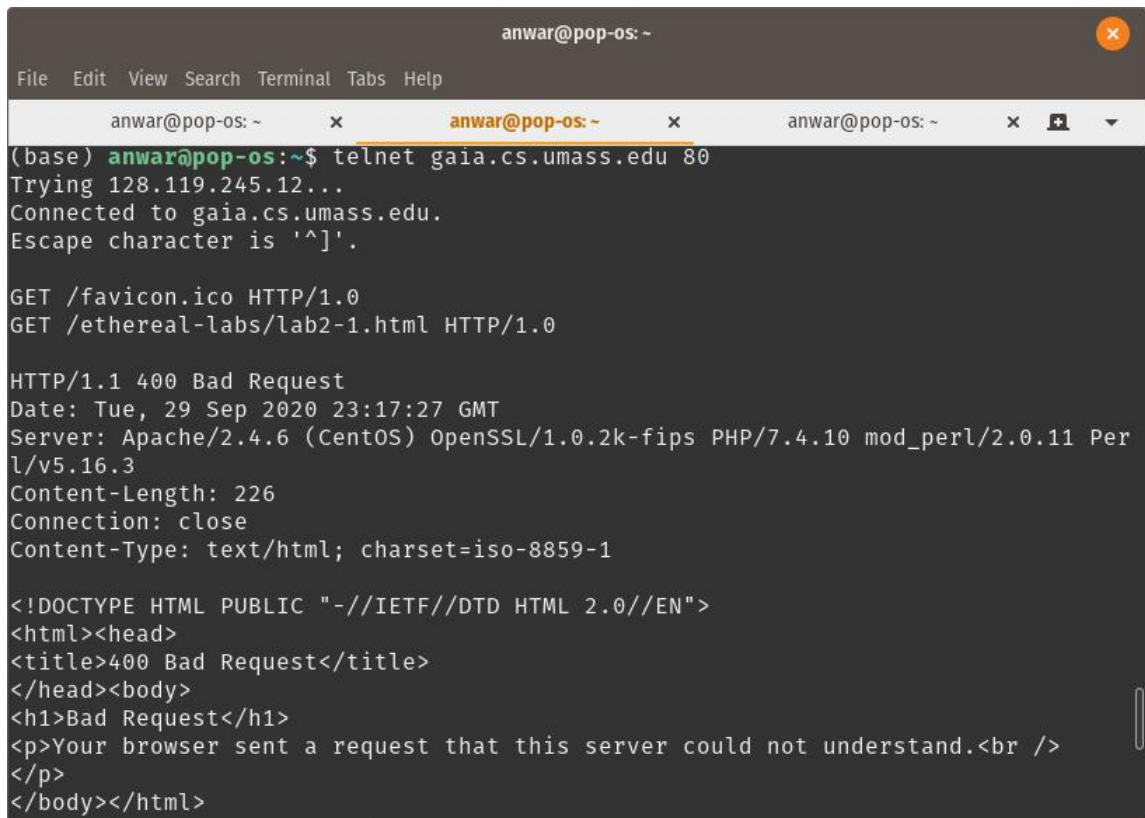
C. Running Telnet

1. Open the terminal (command prompt in Windows) and then type:
telnet gaia.cs.umass.edu 80. (NOTE: gaia.cs.umass.edu is the host and 80 the port where HTTP service runs).
 - a. It'll take some time to establish TCP connection
2. After successful telnet connection, type the following:
 - a. GET /ethereal-labs/lab2-1.html HTTP/1.0 [Enter]
 - b. Host: gaia.cs.umass.edu [Enter 2x]
3. You will see response from the server and then the connection will be immediately closed:



```
anwar@pop-os: ~  
File Edit View Search Terminal Tabs Help  
anwar@pop-os: ~ x anwar@pop-os: ~ x anwar@pop-os: ~ x +  
(base) anwar@pop-os:~$ telnet gaia.cs.umass.edu 80  
Trying 128.119.245.12...  
Connected to gaia.cs.umass.edu.  
Escape character is '^]'.  
GET /ethereal-labs/lab2-1.html HTTP/1.0  
Host: gaia.cs.umass.edu  
HTTP/1.1 408 Request Timeout  
Date: Tue, 29 Sep 2020 23:11:47 GMT  
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.10 mod_perl/2.0.11 Perl/v5.16.3  
Content-Length: 221  
Connection: close  
Content-Type: text/html; charset=iso-8859-1  
  
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">  
<html><head>  
<title>408 Request Timeout</title>  
</head><body>  
<h1>Request Timeout</h1>  
<p>Server timeout waiting for the HTTP request from the client.</p>  
</body></html>  
Connection closed by foreign host.  
(base) anwar@pop-os:~$ telnet gaia.cs.umass.edu 80  
Trying 128.119.245.12...
```

4. Create another telnet connection to gaia.cs.umass.edu web server (as in point 1)
5. After successful telnet connection, type the following:
 - a. GET /favicon.ico HTTP/1.0 [Enter]
 - b. Host: gaia.cs.umass.edu [Enter 2x]
6. Again you will see the response from the server and closing of the connection:



```
anwar@pop-os: ~  
File Edit View Search Terminal Tabs Help  
anwar@pop-os: ~ x anwar@pop-os: ~ x anwar@pop-os: ~ x  
(base) anwar@pop-os:~$ telnet gaia.cs.umass.edu 80  
Trying 128.119.245.12...  
Connected to gaia.cs.umass.edu.  
Escape character is '^]'.  
  
GET /favicon.ico HTTP/1.0  
GET /ethereal-labs/lab2-1.html HTTP/1.0  
  
HTTP/1.1 400 Bad Request  
Date: Tue, 29 Sep 2020 23:17:27 GMT  
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.10 mod_perl/2.0.11 Perl/v5.16.3  
Content-Length: 226  
Connection: close  
Content-Type: text/html; charset=iso-8859-1  
  
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">  
<html><head>  
<title>400 Bad Request</title>  
</head><body>  
<h1>Bad Request</h1>  
<p>Your browser sent a request that this server could not understand.<br />  
</p>  
</body></html>
```

D. Analyze Wireshark packet capture

1. Stop Wireshark packet capture
2. Type “tcp.port == x || tcp.port == y” in the display-filter window (where **x** is the port number of your end device that is involved in the first telnet connection, and **y** is the port number of your end device that is involved in the second telnet connection), so that only TCP messages in both telnet session previously performed will be displayed.
 - a. You can check the port numbers that are involved in the communication with the gaia.cs.umass.edu web server (IP Address 128.119.245.12)
 - b. In the example below, the port number of the client device in the first connection, **x** = 54080, while in the second connection, **y** = 54082.

Wireshark interface showing network traffic on interface wlo1. The packet list displays various protocols including TCP, HTTP, ARP, and MDNS. The packet details pane shows the structure of a selected packet (Frame 1: 66 bytes on wire).

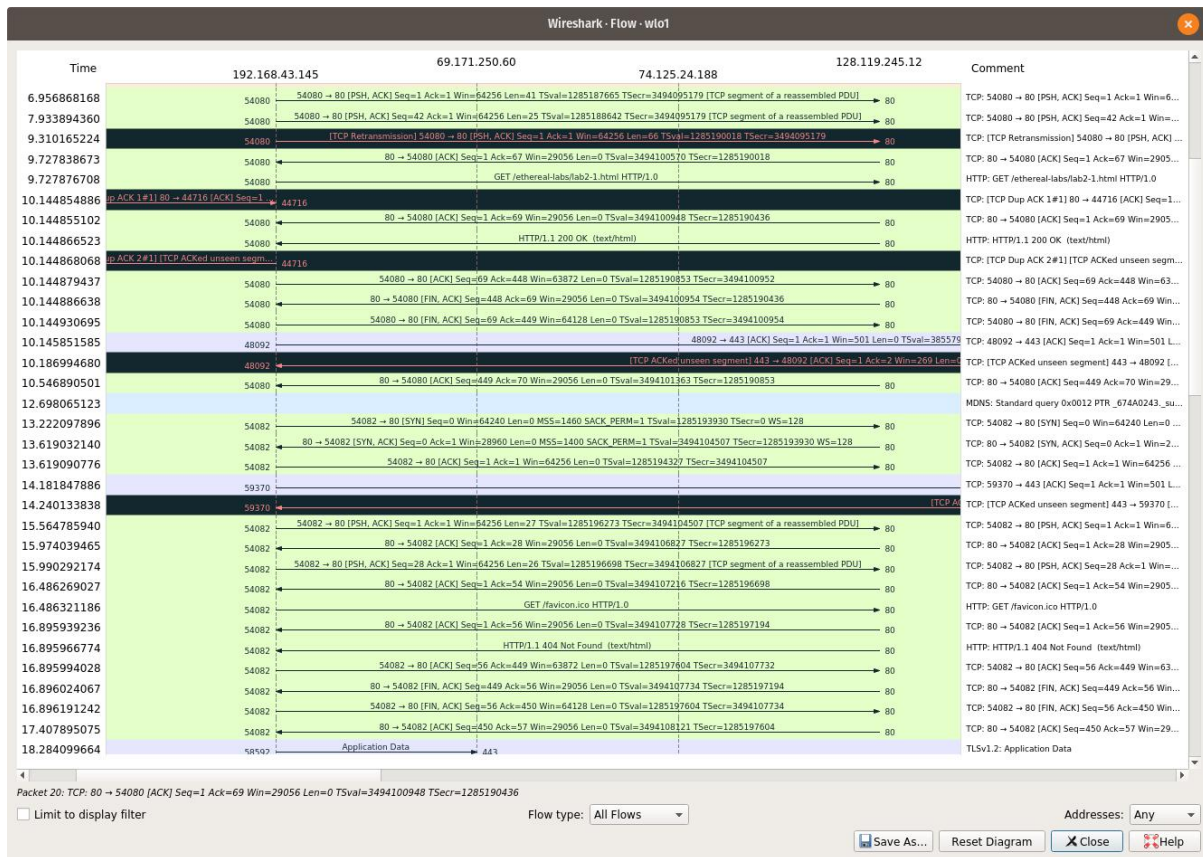
No.	Time	Source	Destination	Protocol	Length	Info
9	3.923523827	192.168.43.145	128.119.245.12	TCP	74	54080 → 80 [SYN] Seq=0 Win=64240 Len=0 ...
10	4.300442616	128.119.245.12	192.168.43.145	TCP	74	80 → 54080 [SYN, ACK] Seq=0 Ack=1 Win=2...
11	4.300517511	192.168.43.145	128.119.245.12	TCP	66	54080 → 80 [ACK] Seq=1 Ack=1 Win=64256 ...
12	4.914785385	0e:a8:a7:45:bd:dc	IntelCor_67:ea:a8	ARP	42	Who has 192.168.43.145? Tell 192.168.43...
13	4.914809710	IntelCor_67:ea:a8	0e:a8:a7:45:bd:dc	ARP	42	192.168.43.145 is at 08:71:90:67:ea:a8
14	6.956868168	192.168.43.145	128.119.245.12	TCP	107	54080 → 80 [PSH, ACK] Seq=1 Ack=1 Win=6...
15	7.933894360	192.168.43.145	128.119.245.12	TCP	91	54080 → 80 [PSH, ACK] Seq=42 Ack=1 Win=...
16	9.310165224	192.168.43.145	128.119.245.12	TCP	132	[TCP Retransmission] 54080 → 80 [PSH, A...
17	9.727838673	128.119.245.12	192.168.43.145	TCP	66	80 → 54080 [ACK] Seq=1 Ack=67 Win=29056...
18	9.727876708	192.168.43.145	128.119.245.12	HTTP	68	GET /etheral-labs/lab2-1.html HTTP/1.0
19	10.144854886	139.99.122.30	192.168.43.145	TCP	66	[TCP Dup ACK 1#1] 80 → 44716 [ACK] Seq=...
20	10.144855102	128.119.245.12	192.168.43.145	TCP	66	80 → 54080 [ACK] Seq=1 Ack=69 Win=29056...
21	10.144866523	128.119.245.12	192.168.43.145	HTTP	513	HTTP/1.1 200 OK (text/html)
22	10.144868068	192.168.43.145	139.99.122.30	TCP	66	[TCP Dup ACK 2#1] [TCP ACKed unseen seq...
23	10.144879437	192.168.43.145	128.119.245.12	TCP	66	54080 → 80 [ACK] Seq=69 Ack=448 Win=638...
24	10.144886638	128.119.245.12	192.168.43.145	TCP	66	80 → 54080 [FIN, ACK] Seq=448 Ack=69 Wi...
25	10.144930695	192.168.43.145	128.119.245.12	TCP	66	54080 → 80 [FIN, ACK] Seq=69 Ack=449 Wi...
26	10.145851585	192.168.43.145	74.125.24.101	TCP	66	48092 → 443 [ACK] Seq=1 Ack=1 Win=501 L...
27	10.186994680	74.125.24.101	192.168.43.145	TCP	66	[TCP ACKed unseen segment] 443 → 48092
28	10.546890501	128.119.245.12	192.168.43.145	TCP	66	80 → 54080 [ACK] Seq=449 Ack=70 Win=290...
29	12.698065123	192.168.43.1	224.0.0.251	MDNS	119	Standard query 0x0012 PTR _674A0243._su...
30	13.222097896	192.168.43.145	128.119.245.12	TCP	74	54082 → 80 [SYN] Seq=0 Win=64240 Len=0 ...
31	13.619032140	128.119.245.12	192.168.43.145	TCP	74	80 → 54082 [SYN, ACK] Seq=0 Ack=1 Win=2...
32	13.619090776	192.168.43.145	128.119.245.12	TCP	66	54082 → 80 [ACK] Seq=1 Ack=1 Win=64256 ...
33	14.181847886	192.168.43.145	74.125.200.94	TCP	66	59370 → 443 [ACK] Seq=1 Ack=1 Win=501 L...
34	14.240133838	74.125.200.94	192.168.43.145	TCP	66	[TCP ACKed unseen segment] 443 → 59370
35	15.564785940	192.168.43.145	128.119.245.12	TCP	93	54082 → 80 [PSH, ACK] Seq=1 Ack=1 Win=6...
36	15.974039465	128.119.245.12	192.168.43.145	TCP	66	80 → 54082 [ACK] Seq=1 Ack=28 Win=29056...
37	15.990292174	192.168.43.145	128.119.245.12	TCP	92	54082 → 80 [PSH, ACK] Seq=28 Ack=1 Win=...
38	16.486269027	128.119.245.12	192.168.43.145	TCP	66	80 → 54082 [ACK] Seq=1 Ack=54 Win=29056...

Frame 1: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface wlo1, id 0
 Ethernet II, Src: 0e:a8:a7:45:bd:dc (0e:a8:a7:45:bd:dc), Dst: IntelCor_67:ea:a8 (08:71:90:67:ea:a8)
 Internet Protocol Version 4, Src: 139.99.122.30, Dst: 192.168.43.145
 Transmission Control Protocol, Src Port: 80, Dst Port: 44716, Seq: 1, Ack: 1, Len: 0

0000 08 71 90 67 ea a8 0e a8 a7 45 bd dc 08 00 45 00 .q.g....E....E.
 0010 00 34 e0 89 40 00 32 06 76 7f 8b 63 7a 1e c0 a8 .4..@.2.v..cz...
 0020 2b 91 00 50 ae ac 34 53 bd c4 89 63 61 d3 80 10 +..P..4S...ca...
 0030 01 f5 d0 bc 00 00 01 01 08 0a 68 68 94 7c 54 b4hh..|T..
 0040 d4 6b .k

wireshark_wlo1_20200930065935_F6VuRr.pcapng Packets: 96 · Displayed: 96 (100.0%) Profile: Default

3. Open the flow graph and then tick the “Limit to display filter” option. You can ignore the TCP retransmission message and the TCP message with “PSH” flag.
 - a. Please compare with the pattern of HTTP non-persistent message flow in the slide!
 - b. Notice the TCP message with flag “SYN” and “FIN”. (You will find out the detail about them in the next chapter).



E. Your Task

- Conduct steps A-D above.
- Try an experiment to access another website by using HTTP, access at least two different links to the website menu/sub-menu/assets. Identify the request line, status line, header lines, dan body/data from the captured packet on wireshark
- Screenshot your entire experiment and save it in a .pdf file.
- Answer question below comprehensively (200 words/more) on your pdf file along with task no.3
 - Explain the flow of HTTP persistent in your experiment
 - Explain the differences of HTTP persistent and HTTP non persistent

F. Submission

- Wireshark files of your experiment
- Pdf file of your screenshot and analysis

* zip the files into one .zip file then upload it to scele.

Reference : Tutorial Wireshark Jarkom A/B/C Gasal 2020_2021