THE UNIVERSITY OF MELBOURNE
MELBOURNE SCHOOL OF ENGINEERING

Engineering Assignment Coversheet

Please note that you:

- Must keep a full copy of your submission for this assignment
- Must staple this assignment
- Must NOT use binders or plastic folders except for large assignments

Student Number(s)

850509
872301

Group Code (if applicable):

Assignment Title:	Workshop 1 - Wireshark
Subject Number:	ELEN90061
Subject Name:	Communication Networks
Student Name:	Yi Jian, Yue Chang
Lecturer/Tutor:	Tansu Alpcan
Due Date:	28 Aug

For Late Assignments Only

Has an extension been granted? Yes / No (circle)

A per-day late penalty may apply if you submit this assignment after the due date/extension. Please check with your Department/coordinator for further information.

Plagiarism

Plagiarism is the act of representing as one's own original work the creative works of another, without appropriate acknowledgment of the author or source.

Collusion

Collusion is the presentation by a student of an assignment as his or her own which is in fact the result in whole or in part of unauthorised collaboration with another person or persons. Collusion involves the cooperation of two or more students in plagiarism or other forms of academic misconduct.

Both collusion and plagiarism can occur in group work. For examples of plagiarism, collusion and academic misconduct in group work please see the University's policy on Academic Honesty and Plagiarism: http://academichonesty.unimelb.edu.au/

Plagiarism and collusion constitute cheating. Disciplinary action will be taken against students who engage in plagiarism and collusion as outlined in University policy. Proven involvement in plagiarism or collusion may be recorded on my academic file in accordance with Statute 13.1.18.

STUDENT DECLARATION

Please sign below to indicate that you understand the following statements:

I declare that:

- This assignment is my own original work, except where I have appropriately cited the original source.
- This assignment has not previously been submitted for assessment in this or any other subject.

For the purposes of assessment, I give the assessor of this assignment the permission to:

- Reproduce this assignment and provide a copy to another member of staff; and
- Take steps to authenticate the assignment, including communicating a copy of this assignment to a checking service (which may retain a copy of the assignment on its database for future plagiarism checking).

Student signature 常悦 .... Date 08/24/2019 ....

Workshop 1

Q1.1

Four main uses of WireShark:

- Network administrators use it to *troubleshoot network problems*
- Network security engineers use it to *examine security problems*
- Developers use it to *debug protocol implementations*
- People use it to *learn network protocol* internals [1]

The last one applies to us, we are using it to learn network protocol internals.

Q1.2

The two security related issues are examine security problems and troubleshoot network problems, because WireShark provide a detailed packet data and it has useful tools such as filters to help with analyze.

Q1.3

In order to capture TokenRing traffic other than Unicast traffic to and from the host on which you're running Wireshark, Multicast traffic, and Broadcast traffic, the adapter will have to be put into promiscuous mode, so that the filter mentioned above is switched off and all packets received are delivered to the host.

The Windows driver for the Madge Presto PCI 2000 TokenRing adapter requires you to enable promiscuous mode explicitly in order to do this, and the drivers for other Madge TokenRing adapters allow promiscuous mode to be disabled, in which case promiscuous mode will have to be re-enabled. Note that those drivers also support permanentlydisabling promiscuous mode; promiscuous mode can never be re-enabled on an adapter on which promiscuous mode has been permanently disabled. [2]

Q1.4

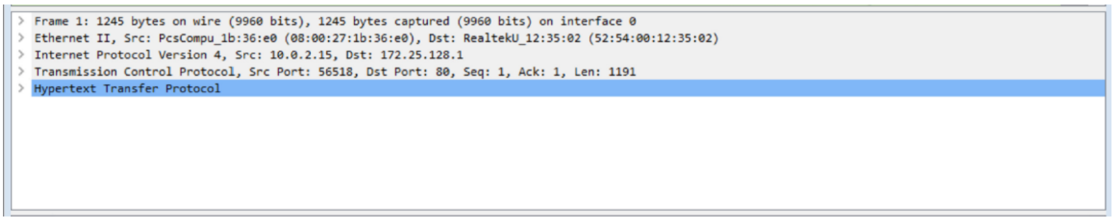
The filter will get the packets send from or to TCP traffic port http.

Q1.5

The three main horizontal content segments are “Packet List” pane, “Packet Details” pane and “Packet Bytes” pane.

- The “Packet List” pane displays all the packets in the current capture file. [3]
- The “Packet Details” pane shows the current packet (selected in the “Packet List” pane) in a more detailed form. [4]
- The “Packet Bytes” pane shows a canonical hex dump of the packet data. [5]

Q1.6



```
> Frame 1: 1245 bytes on wire (9960 bits), 1245 bytes captured (9960 bits) on interface 0
> Ethernet II, Src: PcsCompu_1b:36:e0 (08:00:27:1b:36:e0), Dst: RealtekU_12:35:02 (52:54:00:12:35:02)
> Internet Protocol Version 4, Src: 10.0.2.15, Dst: 172.25.128.1
> Transmission Control Protocol, Src Port: 56518, Dst Port: 80, Seq: 1, Ack: 1, Len: 1191
> Hypertext Transfer Protocol
```

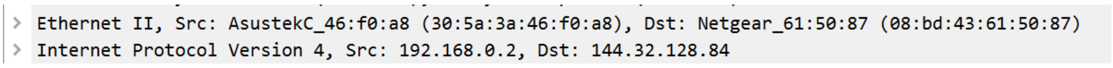
Figure 1 HTTP packet

First line is physical layer, as we can bytes and bits data bits on wire.
Second line is data link layer, as we can see Ethnet
Third line is network layer, as we can see routing from source to destination
Fourth line is transport layer, as we can see TCP protocol.
The last line is application layer, as we can see HTTP.

Q 2.1

Capture filter capture certain type of packets while display filter display certain type of captured packets.

Q2.2



```
> Ethernet II, Src: AsustekC_46:f0:a8 (30:5a:3a:46:f0:a8), Dst: Netgear_61:50:87 (08:bd:43:61:50:87)
> Internet Protocol Version 4, Src: 192.168.0.2, Dst: 144.32.128.84
```

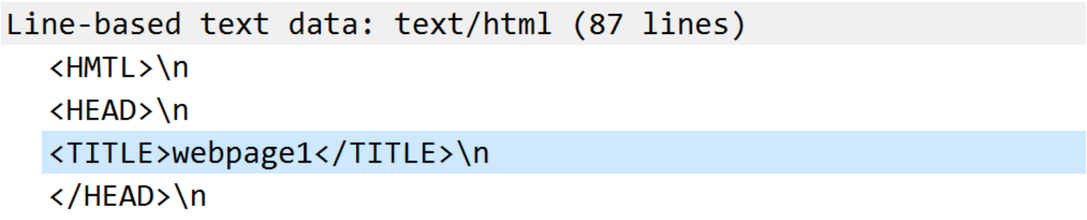
Figure 2 MAC and IP address

Client IP address: 192.168.0.2
Server IP address: 144.32.128.84
Client MAC address: 30:5a:3a:46:f0:a8
Next hop router MAC address: 08:BD:43:61:50:87

Q2.3

The browser send a GET request to the server, the server successfully response to the request by sending back status 200 is OK.

Q2.4



```
Line-based text data: text/html (87 lines)
    <HMTL>\n
    <HEAD>\n
    <TITLE>webpage1</TITLE>\n
    </HEAD>\n
```

Figure 3 web title

The title is webpage1

Q2.5

Gratuitous ARP could mean both gratuitous ARP *request* or gratuitous ARP *reply*. Gratuitous in this case means a request/reply that is not normally needed according to the ARP specification (RFC 826) but could be used in some cases. A gratuitous ARP request is an Address Resolution Protocol request packet where the source and destination IP are both set to the IP of the machine issuing the packet and the destination MAC is the broadcast address ff:ff:ff:ff:ff:ff. Ordinarily, no reply packet will occur. A gratuitous ARP reply is a reply to which no request has been made.

Gratuitous ARPs are useful for four reasons:

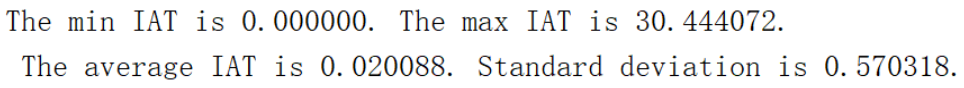
- They can help detect IP conflicts.
- They assist in the updating of other machines' ARP tables.
- They inform switches of the MAC address of the machine on a given switch port, so that the switch knows that it should transmit packets sent to that MAC address on that switch port.
- Every time an IP interface or link goes up, the driver for that interface will typically send a gratuitous ARP to preload the ARP tables of all other local hosts. Thus, a gratuitous ARP will tell us that that host just has had a link up event, such as a link bounce, a machine just being rebooted or the user/sysadmin on that host just configuring the interface up. If we see multiple gratuitous ARPs from the same host frequently, it can be an indication of bad Ethernet hardware/cabling resulting in frequent link bounces. [6]

Q3.1

It has frame number, capture time, source IP and destination IP, protocol type, packet length and information, it doesn't have any packet contents.

Q3.2

We use matlab in this workshop.
We use diff() to calculate the IAT, use min(), max(), mean(), std() to get the statistics about IAT.



```
The min IAT is 0.000000. The max IAT is 30.444072.
 The average IAT is 0.020088. Standard deviation is 0.570318.
```

For instantaneous rate, we count the packets that arrived in each time instance of a second.
For average rate we calculate total arrived packets divided by total time.

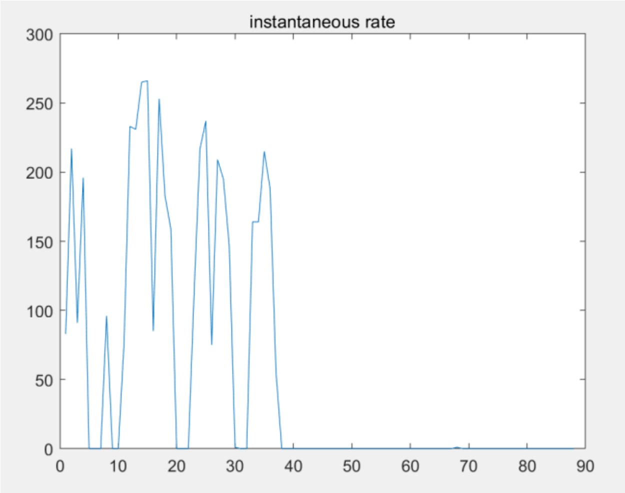


Figure 4 instantaneous rate

```
Average rate is 49.792924.
```

Q3.3

To ‘clean’ the time series, we use prctile(x,p), which return the threshold of the 95th percentile and select the data we need. We use the same matlab function to get statistics of the cleaned data.

```
The minimum of cleaned data is 0.000000.
The maximum of cleaned data is 0.014929.
The average of cleaned data is 0.003775.
The standard deviation of cleaned data is 0.003373.
```

Q3.4

The time series we just plot the cleaned packets IATs.
To plot the histogram of cleaned packets IATs we use matlab function histogram()to plot.

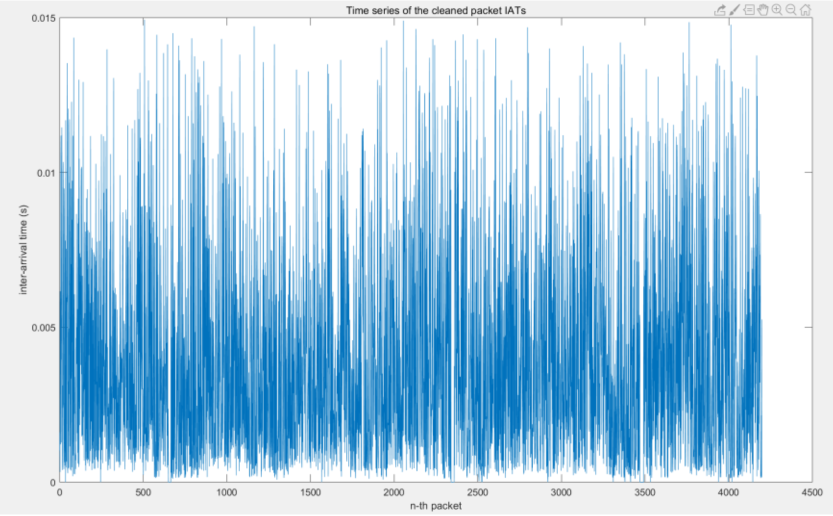


figure 5 time series of the cleaned packet IATs

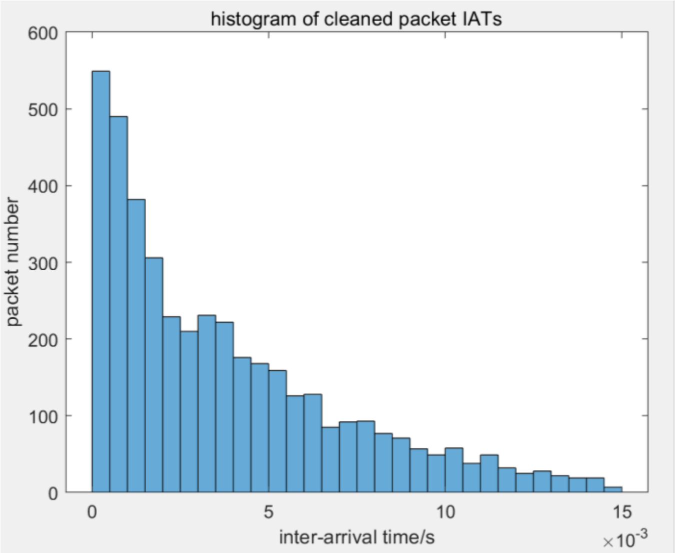


figure 6 histogram of cleaned packet IATs

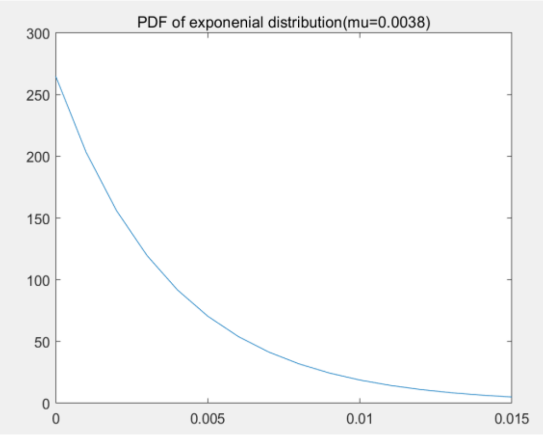
Q3.5

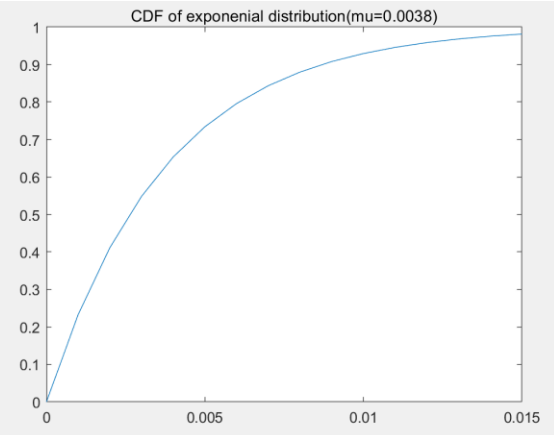
Using expfit in matlab the estimated the mean of an exponentially distributed sample data is 0.0038 which is very close to our cleaned IATs' mean. Mu is 0.0038, lamda = 1/0.0038 so PDF and CDF of exp distribution is:

$$f(x;\lambda) = \begin{cases} \lambda e^{-\lambda x} & , \ x \geq 0, \\ 0 & , \ x < 0. \end{cases}$$

$$F(x;\lambda) = \begin{cases} 1 - e^{-\lambda x} & , \ x \geq 0, \\ 0 & , \ x < 0. \end{cases}$$

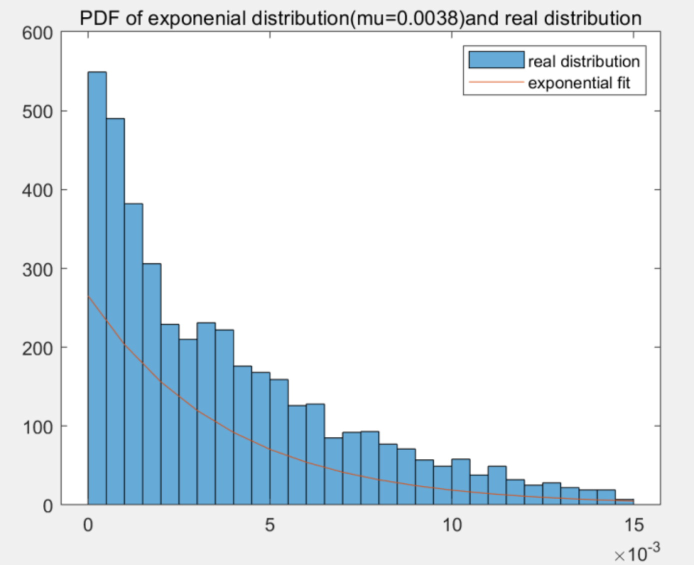
We plot the curve-fitted CDF and PDF:





Q3.6

No. we don't think the Poisson process can accurately model the packet arrival time series. Because Poisson process assume each arrival is independent to each other. In our video streaming case, the packets are not always independent to each other, we can also see from the plot the real distribution is not exact a exponential distribution, it only shows the arrival time interval have an exponential trend, but the exponential fit is not an accurate model for it. So we can only assume the packet arrival is a poisson process, but it can’t accurately model for packet arrival, since each arrival is not independent to each other.



Reference

[1] wireshark. [Online]. Available: https://www.wireshark.org/docs/wsug_html_chunked/ChapterIntroduction.html#ChIntroPurposes.

[2] wireshark. [Online]. Available: https://wiki.wireshark.org/CaptureSetup/TokenRing.

[3] wireshark. [Online]. Available: https://www.wireshark.org/docs/wsug_html_chunked/ChUsePacketListPaneSection.html.

[4] wireshark. [Online]. Available: https://www.wireshark.org/docs/wsug_html_chunked/ChUsePacketDetailsPaneSection.html.

[5] wireshark. [Online]. Available: https://www.wireshark.org/docs/wsug_html_chunked/ChUsePacketBytesPaneSection.html.

[6] wireshark. [Online]. Available: https://wiki.wireshark.org/Gratuitous_ARP.

[7] wireshark. [Online]. Available: https://wiki.wireshark.org/CaptureSetup/Ethernet.