WiresharkLab\_TCP\_UDP

建議:把比例放到150%圖才會清楚

組別:A10

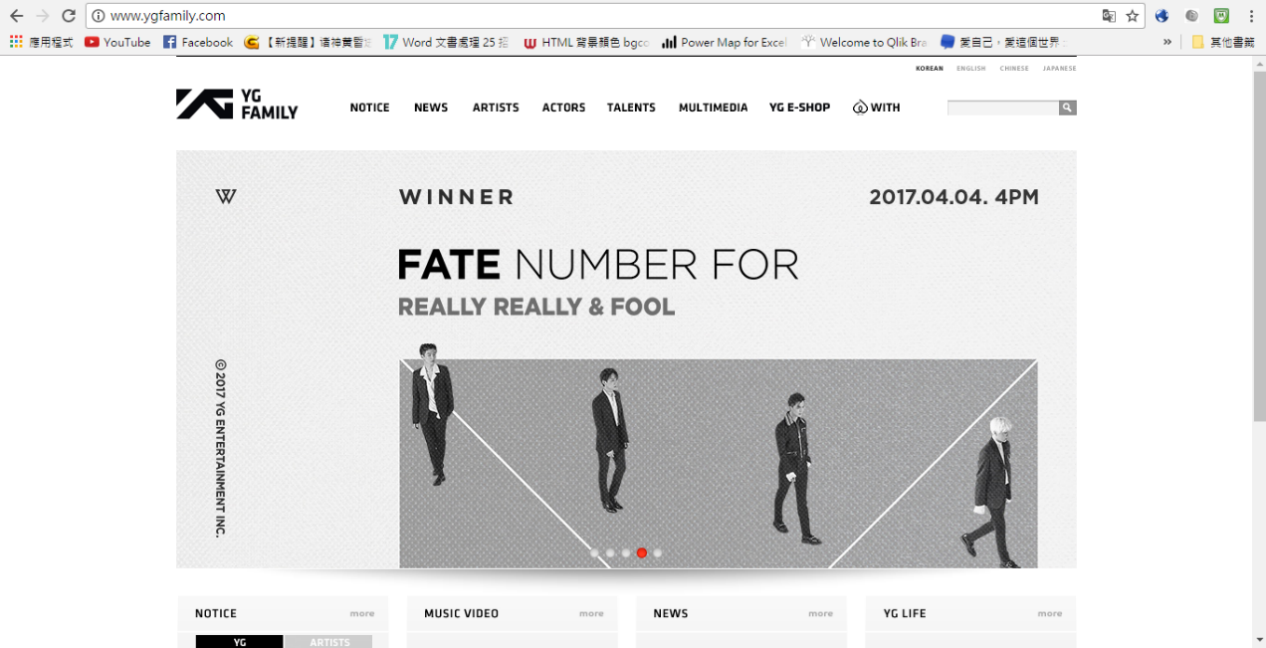
組長:資管二甲 B0444132 張仁樵

組員:資管二甲 B0444101 李積舜

組員:資管二甲 B0444138 潘楚華

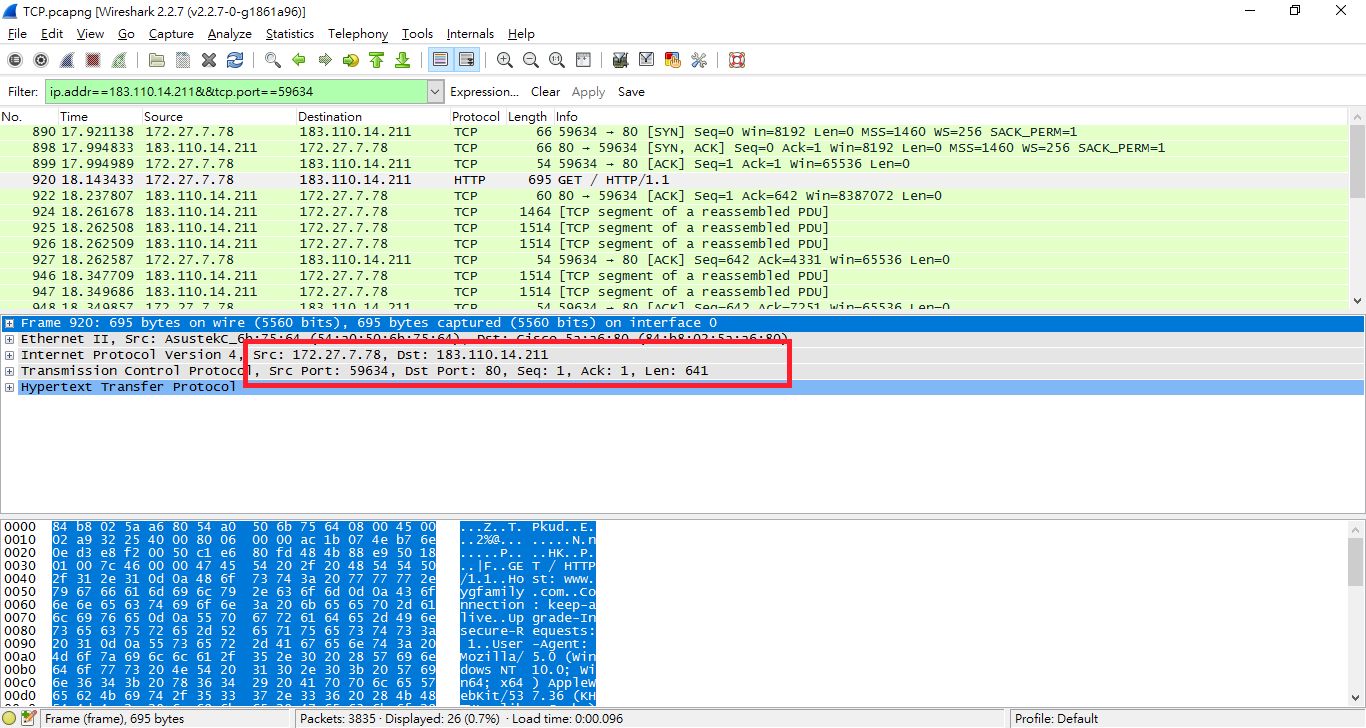
組員:資管二甲 B0444146 林子由

一、TCP



->首先我們進入YG公司(韓國)的官方網站

1.指出Client端與Server端之IP位址和Port號碼。



Client端：

* IP：172.27.7.78
* Port：59634

Server端：

* IP：183.110.14.211
* Port：80

2. 分析其Three-way Handshake階段，說明旗標值、雙方的初始序號 (Initial Sequence Number, ISN)，以及相對應的Acknowledgement Number之變化。

旗標值種類:

1.URG(Urgent data):為1表示高優先級封包，緊急指標欄位有效。

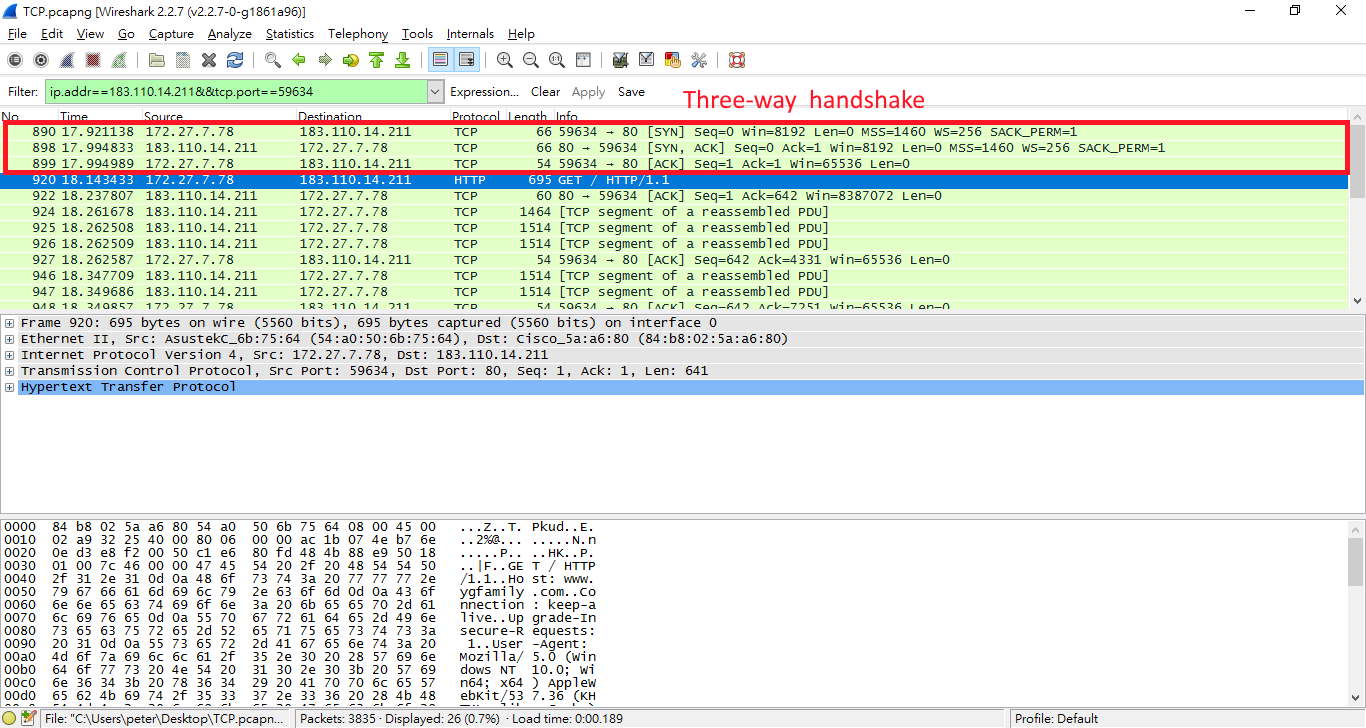
2. ACK(Acknowledge field significant): 為1表示確認號欄位有效，只有第一 個封包沒有設。

3. PSH(Push function): 為1表示是帶有PUSH標誌的資料，指示接收方應該 儘快將這個segment交給應用層而不用等待緩衝區裝滿。

4. RST(Reset): 為1表示出現嚴重差錯，連線會被馬上結束，而無需等待終 止確認手續，還可以用於拒絕非法的segment和拒絕連線請求。

5.SYN(Synchronize sequence number): 為1表示這是連線請求或是連線接受請 求，用於建立連線和使順序號同步。

6.FIN(No more data fro sender, finish): 為1表示傳送方沒有資料要傳輸了，要 求釋放連線。



->紅色框框區域為Three-way handshake的三個封包。

1) Client端先傳送一個SYN flag要求建立連線

(Sequence Number=0, Acknowledgement Number=0)。

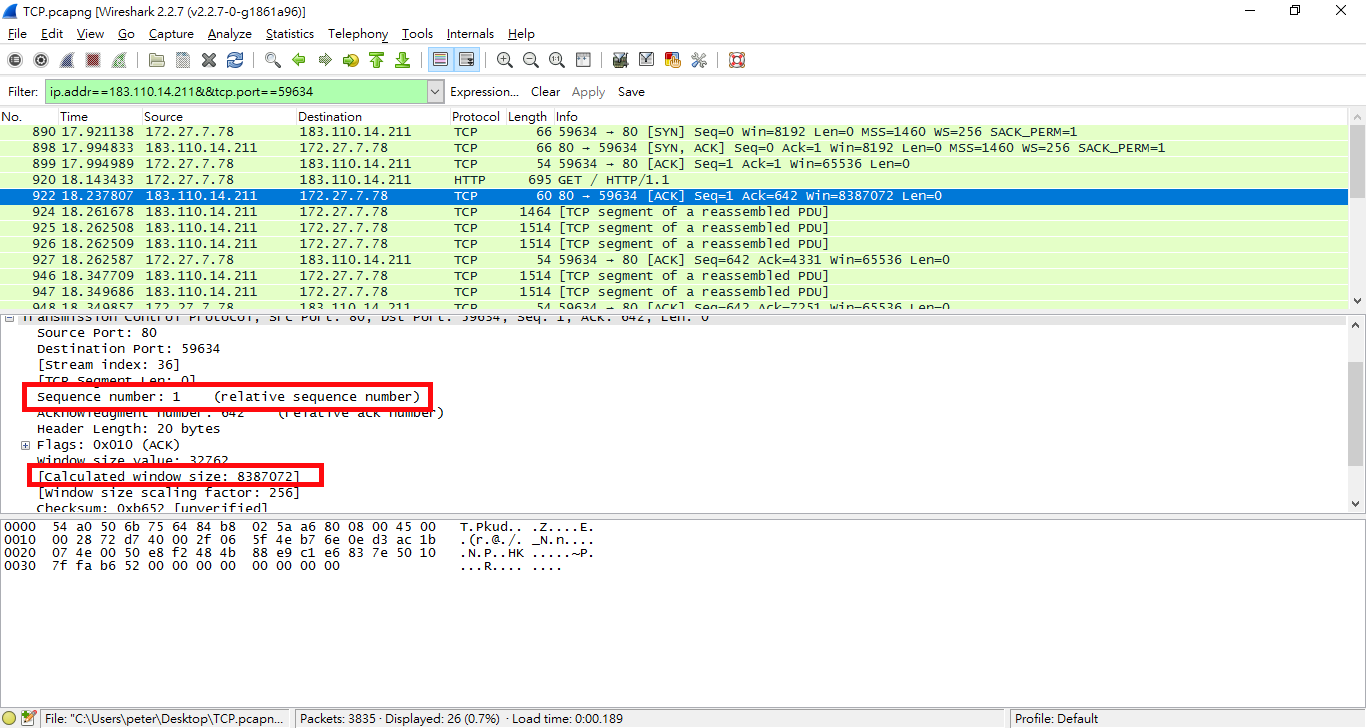
2) Server端收到、同意後會回傳一個SYN flag跟ACK flag

(Sequence Number=0, Acknowledgement Number=1)。

3) Client端收到回應後，再回傳一個ACK flag確認連線建立，開始傳送資料

(Sequence Number=1, Acknowledgement Number=1)。

3.說明真正傳送第一筆資料的序號 (Sequence Number)，以及該封包的Receive Window值大小。

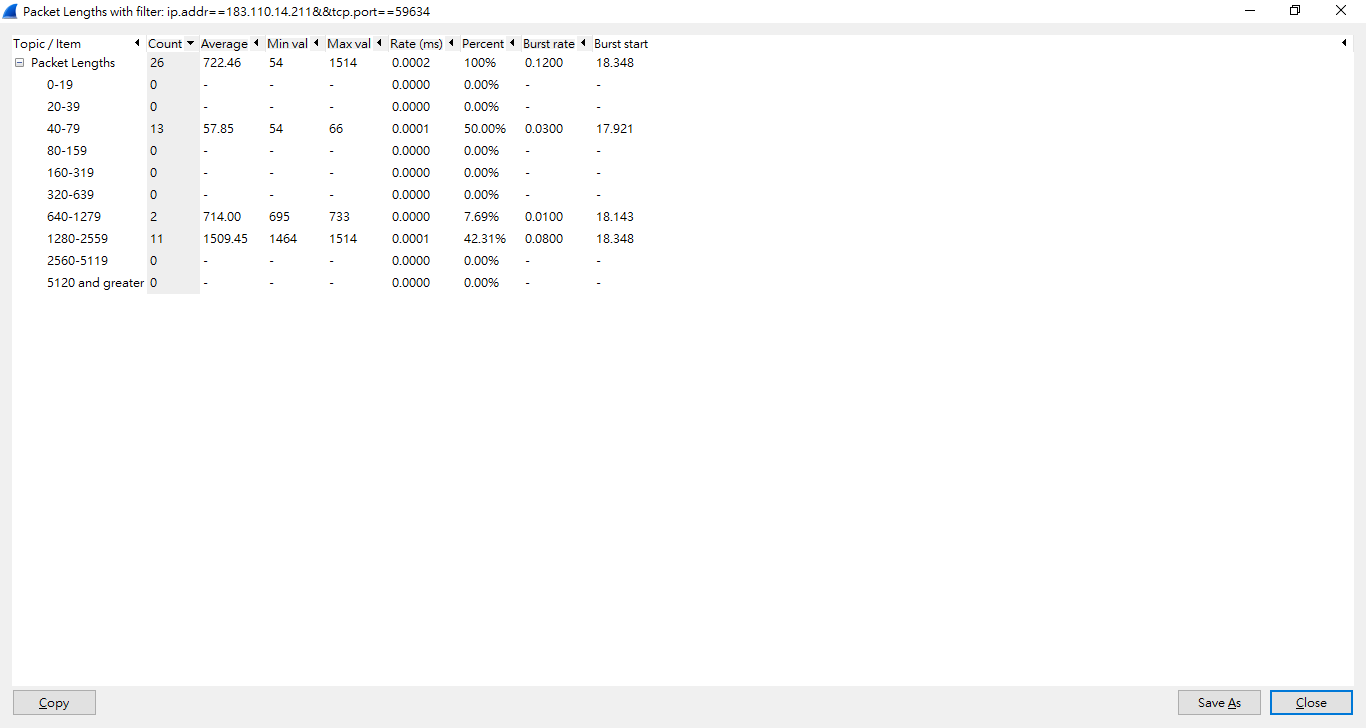


->NO.922封包為真正傳送第一筆資料的封包。

**Sequence number=1**

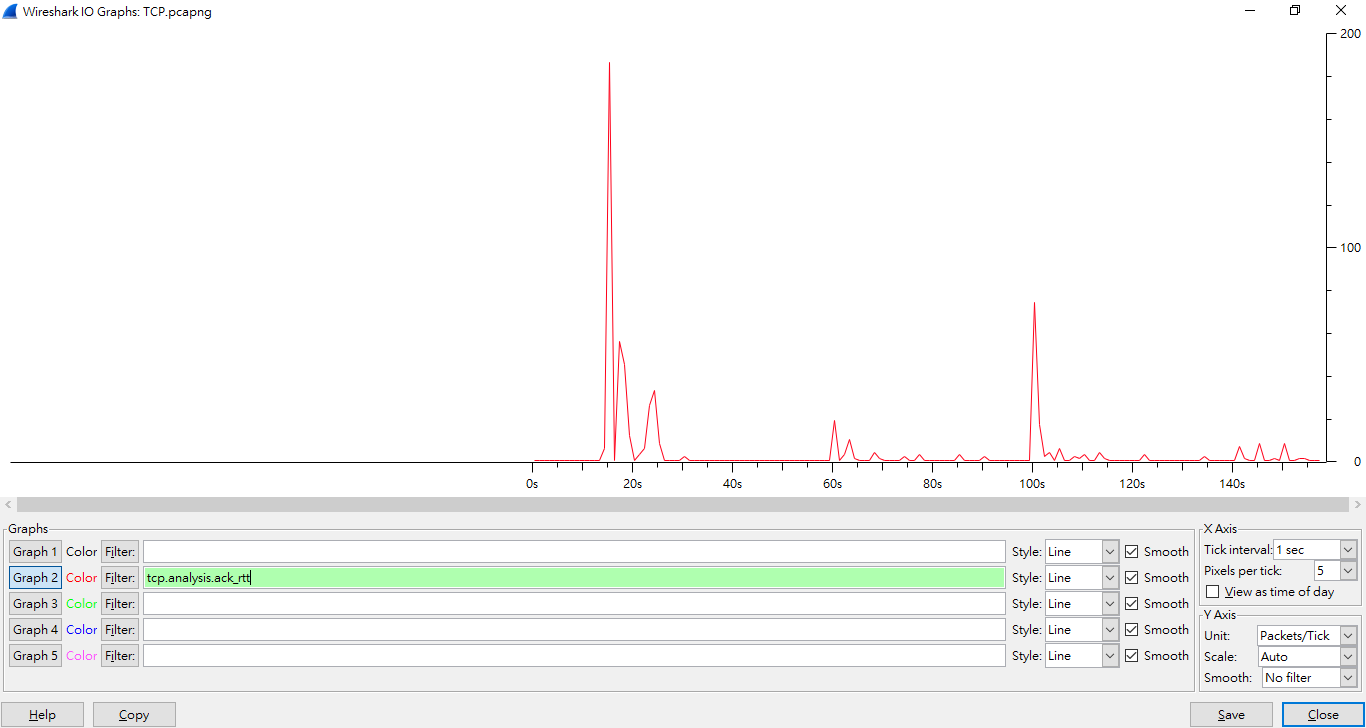
**Receive Window=8387072**

4.分析大致上封包長度的變化。

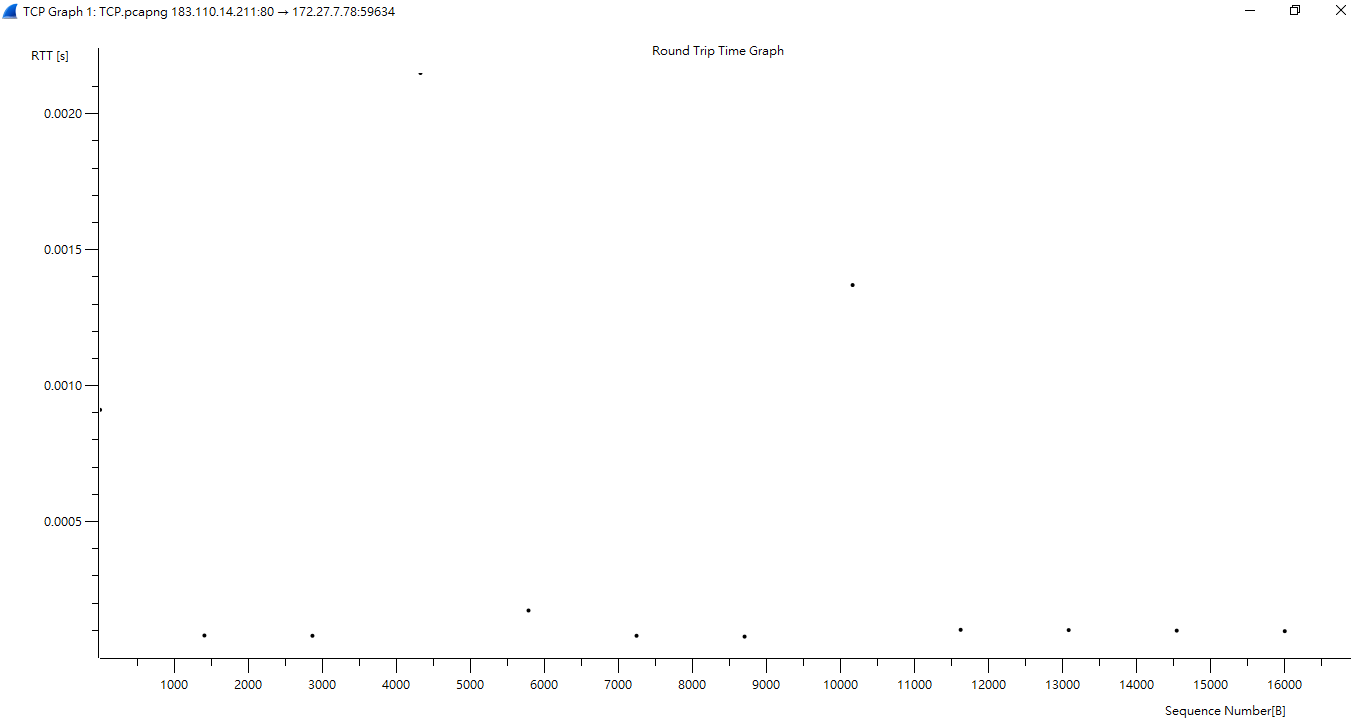


->封包長度有一半都在40-79，至於其他的有2個是屬於HTTP請求的，剩下的13個都是TCP segment of a reassembled PDU。

5. 分析Round Trip Time (RTT) 值的變化。

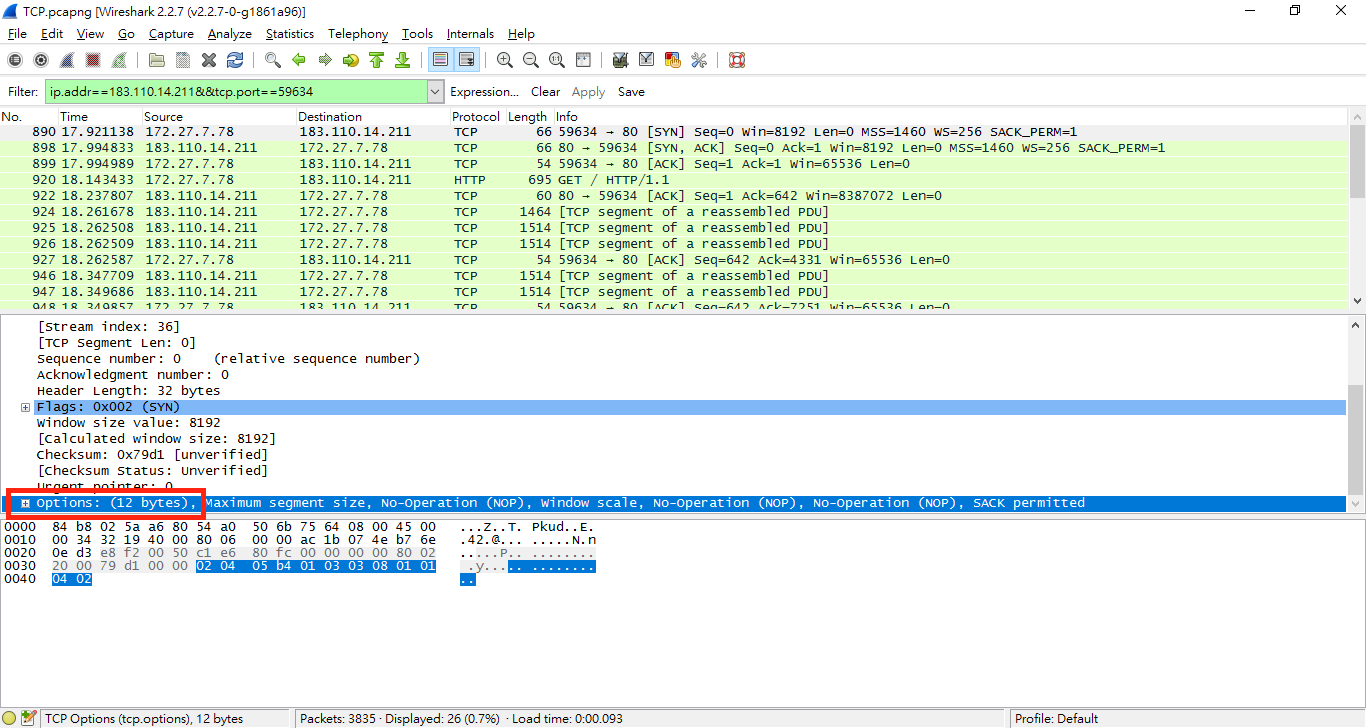


->上圖為利用tcp.analysis.ack\_rtt來表示擷取封包時RTT的變化。



->上圖為RTT Graph顯示RTT的分布，我們可以看到RTT的時間都壓在0.0015秒以下。

6. 說明TCP標頭中是否有Options的欄位。



->在這次擷取封包中，只有Three-way handshake的前兩次有Option，因為有傳

SYN flag(同步)，其餘的TCP 標頭都沒有這項欄位。

這個選項比較少用。當那些需要同步動作的程式，要處理好終端的交互模式，就會使用到 option 來指定資料封包的大小，或會被使用在傳送端和接收端協調最大區段大小(MSS)時，或者在高速網路中用作窗格的縮放係數(window scaling factor)，此外TCP也定義了時間戳記(time-stamping)選項。Option 的長度要麼是 0 ，要麼就是 32bit 的整倍數，最多40bytes，每個Option的開始是1byte的kind欄位，即使資料不足數，也要使用表頭中沒有的資料填夠。

Options類型:

0: 選項表結束（1byte）。

1: 無操作（1byte）用於選項欄位之間的字邊界對齊。

2: 最大segment長度（4bytes，Maximum Segment Size，MSS）通常在建立 連線而設定SYN標誌的封包中指明這個選項，指明本端所能接收的最大長 度的segment。通常將MSS設定為（MTU-40）bytes，攜帶TCP segment的 IP資料報的長度就不會超過MTU，從而避免本機發生IP分片。只能出現 在同步segment中，否則將被忽略。

3: 窗口擴大因子（4bytes，window scale），取值0-14。用來把TCP的窗口的值 左移的位數。只能出現在同步segment中，否則將被忽略。這是因為現在的 TCP接收資料緩衝區（接收窗口）的長度通常大於65535bytes。

4: sackOK—傳送端支援並同意使用SACK選項。

5: SACK實際工作的選項。

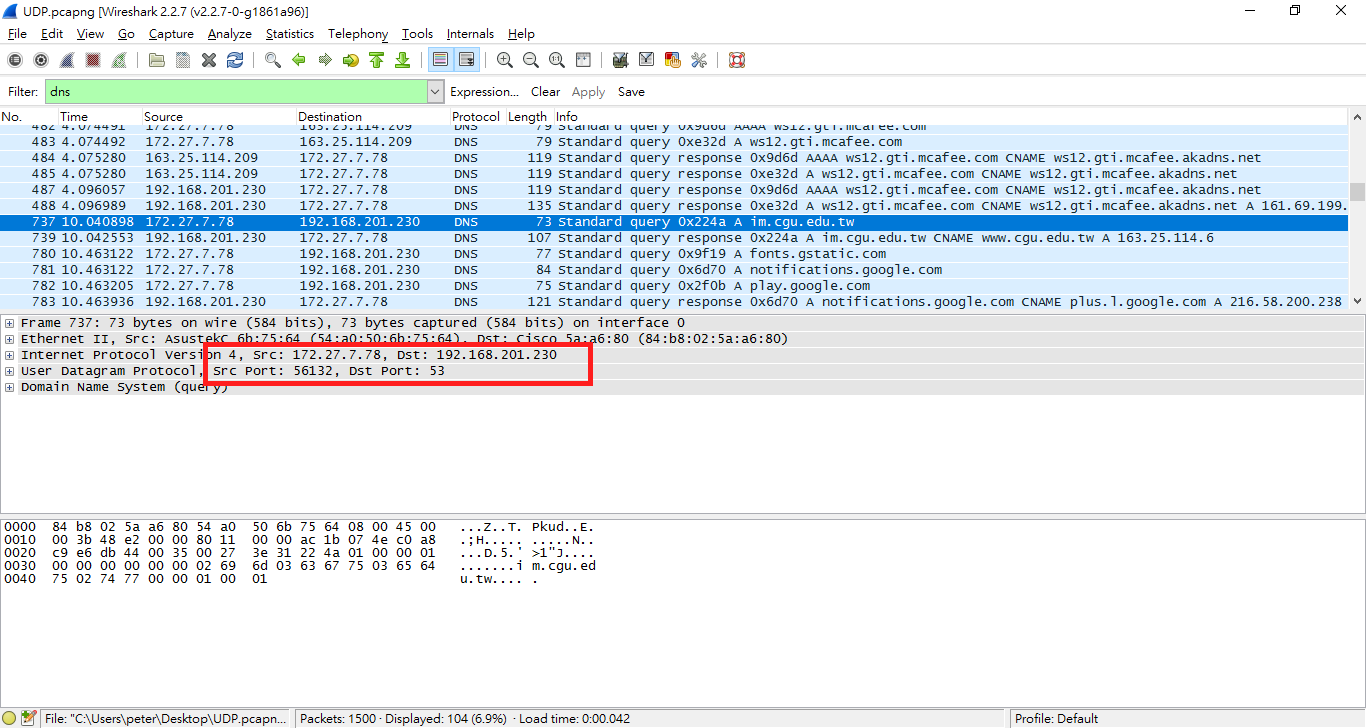
8: 時間戳（10bytes，TCP Timestamps Option，TSopt）。

二、UDP



->首先我們先進入系上首頁

1. 指出Client端與Server端之IP位址和Port號碼。



->NO.737封包為系上網站的封包。

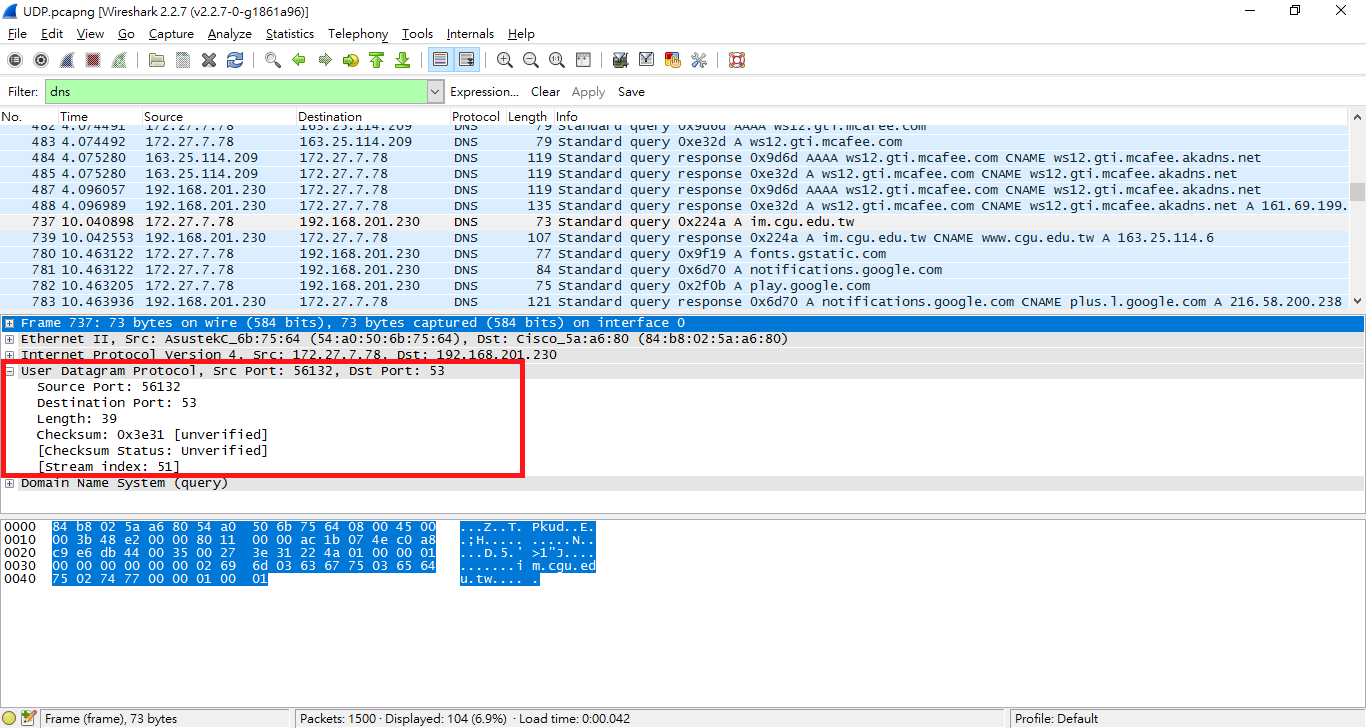
Client端：

* IP：172.27.7.78
* Port：56132

Server端：

* IP：192.168.201.230
* Port：53

2. 任意選取一個封包，說明UDP標頭的各個欄位。



->紅色框框為UDP標頭

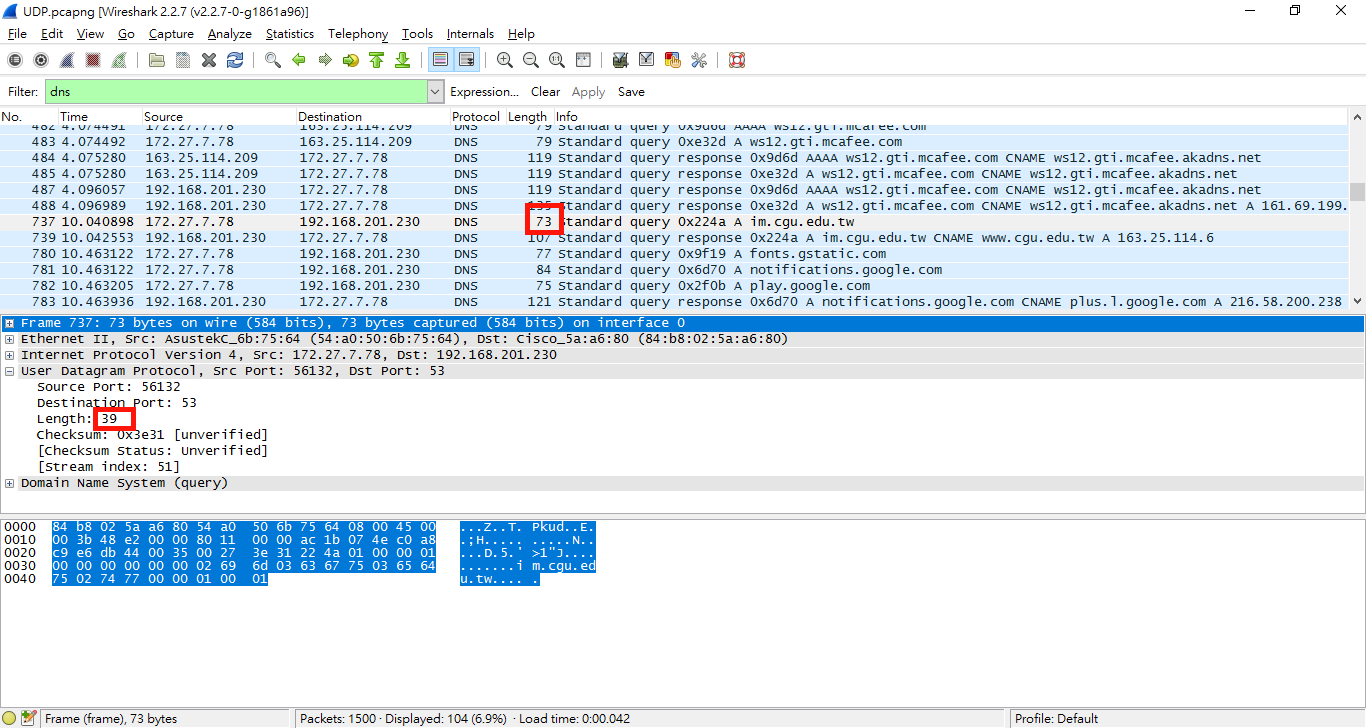
1.Source Port & Destination Port: 跟 TCP 的 Port 一樣，就是 Socket Pair 中的兩個元素，只是給那些透過 UDP 傳送資料的程式使用而已。

2.Length: 整個 UDP 封包的長度，以bytes為單位( byte )，最小值為 8 。

39-8=31(bytes)為應用程式資料(訊息)(data)的長度。

3.checksum: 提供了錯誤偵測的功能，也就是說，checksum會被用來判斷UDP segment在從來源端移動到目的端時，裡面的位元是否遭到更改(比方說，因為連結上的雜訊，或是在儲存於路由器時遭更改。

3. 說明UDP標頭中Length欄位與封包真正長度之關係。



73=(31+8)+20+(6+6+2)=39+20+14=UDP封包+網路層標頭+連接層標頭

31+8: data實際長度+UDP標頭長度

20: IPV4標頭長度

6+6+2: Source address + Destination address + Type field

數字單位皆為(bytes)

三、心得

這次的實驗，主要讓我們明白TCP和UDP這兩個傳送層協定的不同。

一個是連線導向的可靠傳輸，穩定度及可靠性高－－TCP；一個是非連線型的非可靠傳輸，缺少機制保證資料是否被正確接收，或是否遺失資料，其資料接收也不必按順序進行，且無回傳機制控制資料流速度－－UDP。雖然UPD相較TCP，不穩定，但UDP卻擁有傳輸量大且傳輸迅速的特點。

TCP的穩定，建立在其繁多的機制上，而UDP不需要可靠傳輸，因此不用像TCP一樣需要那麼多封包表頭，其省略的空間將可做為更大的傳輸空間，讓UDP犧牲可靠性，換取速度及空間，而若需要在UDP上作可靠傳輸，確認機制就需要於應用層、應用程式本身提供。

魚與熊掌不可兼得，穩定與速度，無法在傳輸上同時兼具，雖然UDP提供快速便利的傳輸，但在網路傳送上，還是趨向於穩定居多，因此我們常用的ftp、http協定皆屬於TCP的範疇。原來，在傳輸層，有那麼多的機制規劃我們傳送的資料，小小的封包裡隱藏著如此之多的內容，不僅僅是只有我們需要傳送的資料而已，TCP眾多header確保了資料的穩定和安全，讓我們有更順暢可靠的網路使用體驗。