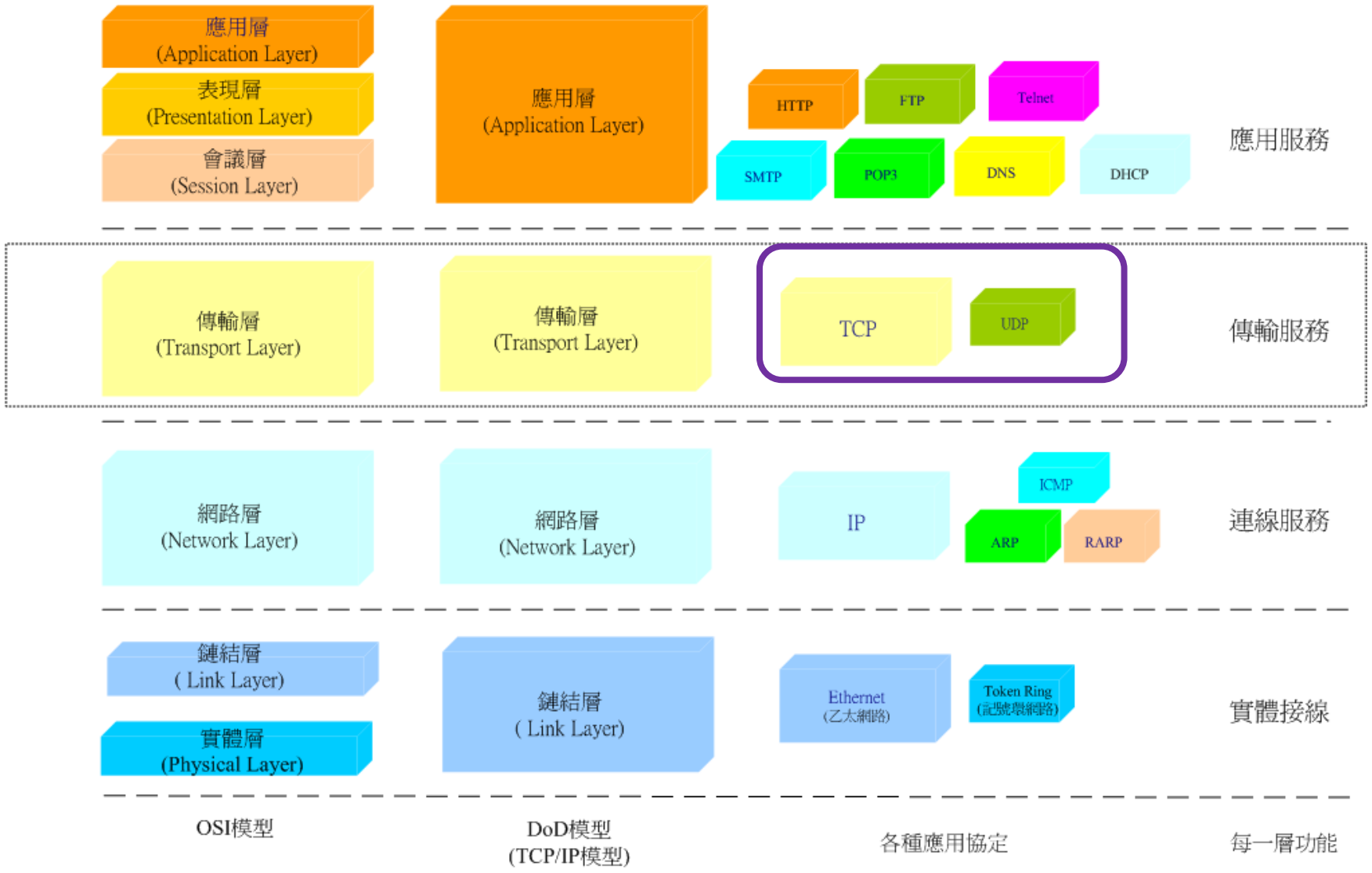
傳輸層 :TCP

TRANSMISSION CONTROL PROTOCOL，傳輸控制協定


應用層
(Application Layer)
表現層
(Presentation Layer)
會議層
(Session Layer)
應用層
(Application Layer)
HTTP
FTP
Telnet
SMTP
POP3
DNS
DHCP
應用服務
傳輸層
(Transport Layer)
傳輸層
(Transport Layer)
TCP
UDP
傳輸服務
網路層
(Network Layer)
網路層
(Network Layer)
IP
ICMP
ARP
RARP
連線服務
鏈結層
(Link Layer)
實體層
(Physical Layer)
鏈結層
(Link Layer)
Ethernet
(乙太網路)
Token Ring
實體接線
OSI模型
DoD模型
(TCP/IP模型)
各種應用協定
每一層功能

TCP/IP通信協定集

- TCP(Transmission Control Protocol)是傳輸控制協定，主要的工作是確保資料正確送達接收端，負責循序編碼和檢查錯誤等。

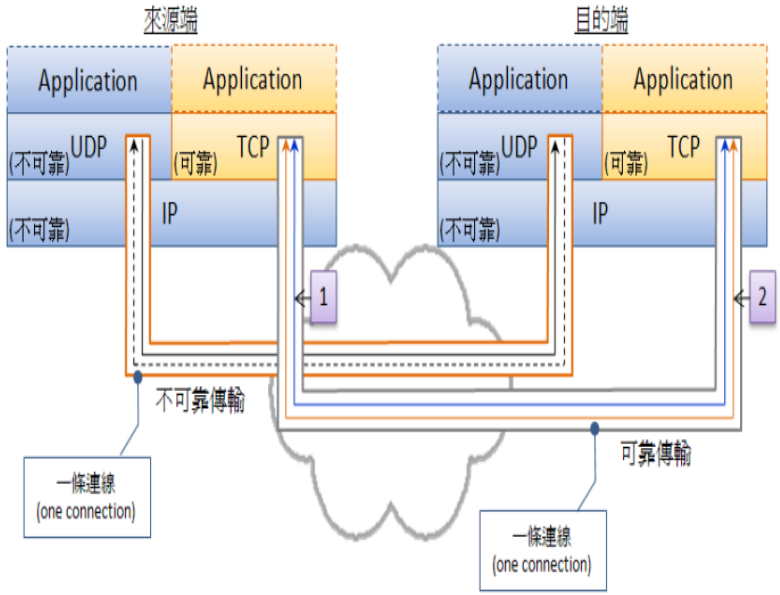
- IP(Internet Protocol)指網際網路協定，負責定義封包的格式、辨識目的地、路徑選擇、傳遞封包等。

傳輸層的連線(connection)

- 透過"IP位址"與"TCP Port"形成一個"Socket"，透過此Socket與對方的Socket形成一個"Socket Pair"進行通訊，也就是一條的"連線"

- "半雙工"(half-duplex)是指兩邊可以雙向傳輸，但是在同一時間僅能有一方傳輸。EX.UDP

- "全雙工"(full-duplex)不僅是兩邊可以雙向傳輸，而且在同一時間，雙方可同時傳輸 。 EX.TCP



TCP主要的功能

- **循序編號(Sequence Number)**
  - TCP為每一個封包建立編號，使封包就算不能按照原來的發送順序抵達接收端，也可依此編號正確重組。
- **確認(Acknowledgement)**
  - 接收端針對發送端所傳來的每一封包，回送「我已收到」的確認封包，類似郵局的雙掛號信中的「回條」觀念。
- **錯誤檢查(Checksum)**
  - TCP在每個封包的表頭中加上一個檢查欄位，以確認其是否為欲傳輸的原始封包。
- **重送(Retransmission)**
  - 發送端如果在某一預定的時間內沒有收到該確認封包，就會認定封包傳輸失敗，於是重送該封包，直到收到該封包抵達接收端的確認封包為止。

TCP的特性

- IP與UDP是以效率為主要訴求，而TCP則是以正確性為主要訴求
  - TCP的特性剛好與IP與UDP的特性完全相反
- 以下將介紹TCP的五項特性
  1. 連線導向的(Connection Oriented)傳輸協定
  2. 同步傳輸(Synchronous Transmission)
  3. 可靠的(Reliable)傳輸協定
  4. 較無效率的(Inefficient)傳輸協定
  5. 流量控制(Flow Control)

1.連線導向的傳輸協定

所謂"連線導向的"(Connection Oriented)傳輸協定，是指在封包傳送之前，必須先經過與對方建立連線之動作，當建立連線成功後，才能開始傳送資料給對方，等資料傳送完畢後，在關閉連線的一種傳輸方式

例如平時我們生活中所使用的電話，若是要與對方通話之前，必須先達成連線後，才能彼此進行通話，此屬於"連線導向"(Connection Oriented)的傳輸協定

2.同步傳輸

所謂"同步傳輸"，是指"傳送端"在傳送出封包後，會等待"接受端"的"確認"(Acknowledge)回應，再傳送下一個 封包，而不是直接將封包不斷地傳送出去。

TCP傳輸協定就是利用同步傳輸來達到彼此的協議，包括

- 確認封包的正確性
- 傳輸速率的控制

3.可靠的傳輸協定

由於TCP協定是屬於“同步傳輸”的一種協定，也是屬於雙向資料傳輸協定，會等待對方確認是否已正確收到，所以TCP封包在網路的轉送的過程，不會因為某些因素而造成封包遺失而不知

縱使封包遺失未到達目的地，或是封包被毀損，來源端都會重新傳送一次該封包 因此TCP協定是一種"可靠的傳輸協定"，也是一個複雜的傳輸協定

4.較無效率的傳輸協定

由於TCP協定是屬於"連線導向"、"同步通訊"及"可靠的"協定也就多出了很多封包傳輸前的連線動作，與傳輸中多了等待對方的"確認"(Acknowledge)回應

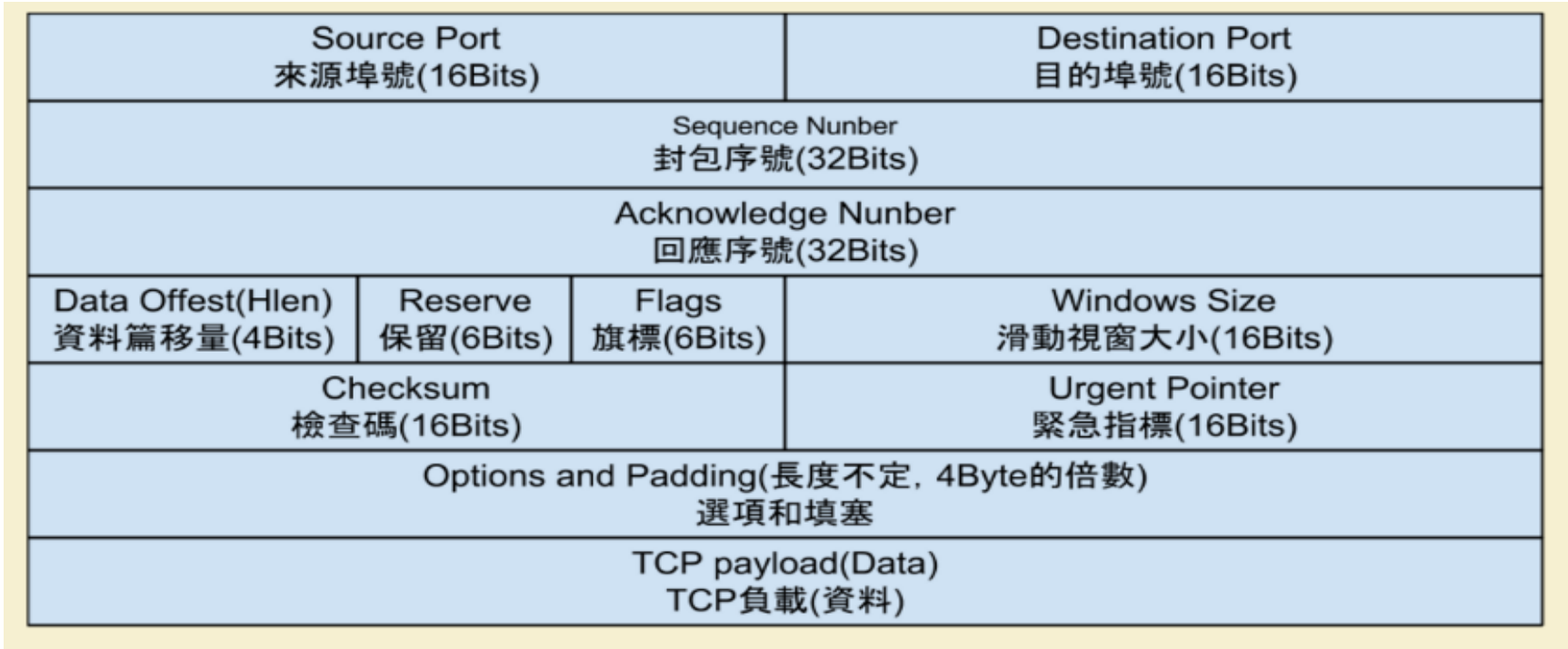
因此在傳輸的整體過程中會較沒有效率，但卻能保證將TCP封包正確無誤地傳達對方

5.流量控制

網際網路無遠弗界、不分國度，每一個使用者電腦的軟、硬體設備有可能都不一樣，當通訊的雙方設備等級差異性太大 時，傳輸過程中，很容易造成處理速度較慢的接受方，因為來不及處理瞬間傳入的資料，而造成資料遺失，並要求另一 方重新傳送一次，無形中已造成網路的沉重負擔 TCP具有流量控制功能，協調彼此雙方都能接受的傳送速度，避免掉因為大量封包湧入，造成主機無法處理的情形下丟棄 (discard)封包，又要求來源端主機重新傳送一次的情形

TCP的封包格式

TCP


Source Port
來源埠號(16Bits)
Destination Port
目的埠號(16Bits)
Sequence Nunber
封包序號(32Bits)
Acknowledge Nunber
回應序號(32Bits)
Data Offest(Hlen)
資料篇移量(4Bits)
Reserve
保留(6Bits)
Flags
旗標(6Bits)
Windows Size
滑動視窗大小(16Bits)
Checksum
檢查碼(16Bits)
Urgent Pointer
緊急指標(16Bits)
Options and Padding(長度不定，4Byte的倍數)
選項和填塞
TCP payload(Data)
TCP負載(資料)

Source Port(16 bits)，來源埠號

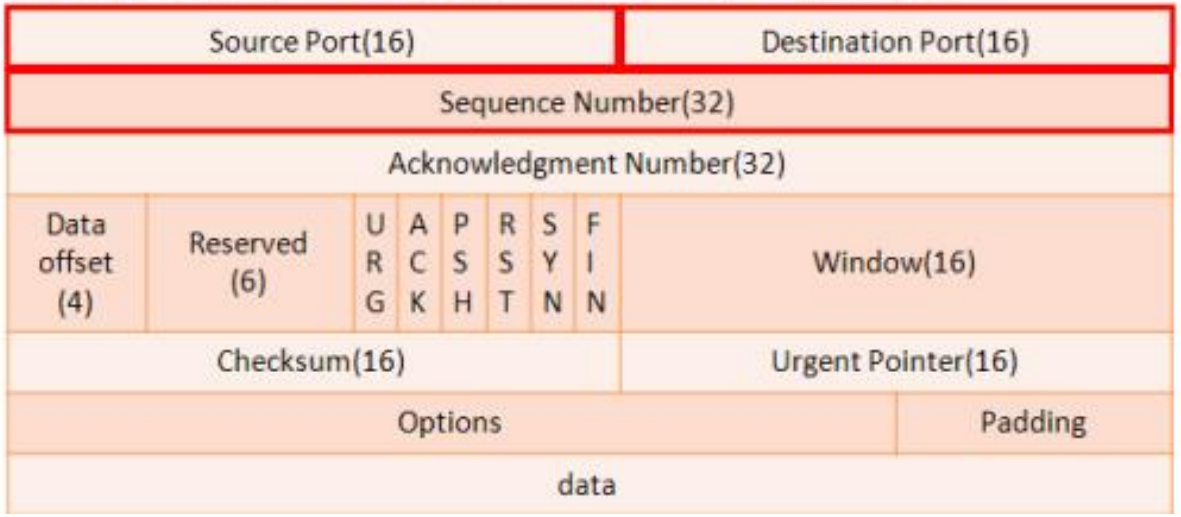
- 長度為16 bits，用來記錄來源主機的埠號；也就是，來源端TCP上層的應用程式，所佔用的TCP埠號

Destination Port(16 bits)，目的埠號

- 長度為16 bits，用來記錄目的主機的埠號；也就是，目的端TCP上層的應用程式，所佔用的TCP埠號

Sequence Number(32 bits)，序號

- 長度為32 bits，序號，也就是為了達到彼此雙方同步的一個號碼，也就是前面介紹中所提到的SEQ_1與SEQ_2的號碼，也代表著『位元組串流』(Byte Stream)中的編號



Acknowledgment Number(32 bits)，確認號碼

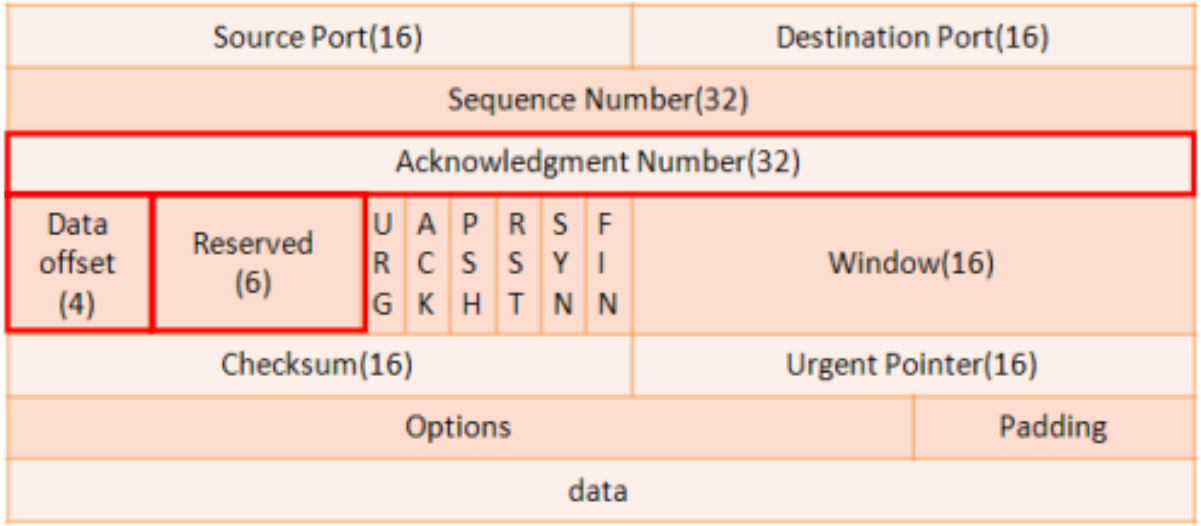
- 長度為32 bits，確認訊息的號碼，此處的號碼會與前面的序號會有關係，也就是紀錄下一次對方傳送過來的序號號碼，亦或是接受資料端告知傳送端，在此編號之前的所有『位元組串流』(Byte Stream)皆已收到

Data offset(4 bits)，資料位移量

- 此欄位長度為4 bits，此處的Data是指TCP封包所承載的資料，所以此處所紀錄的是Data從TCP封包最前面開始的位移量。換言之，就是紀錄TCP Header的長度

Reserved(6 bits)，保留

- 長度為6 bits，保留位元

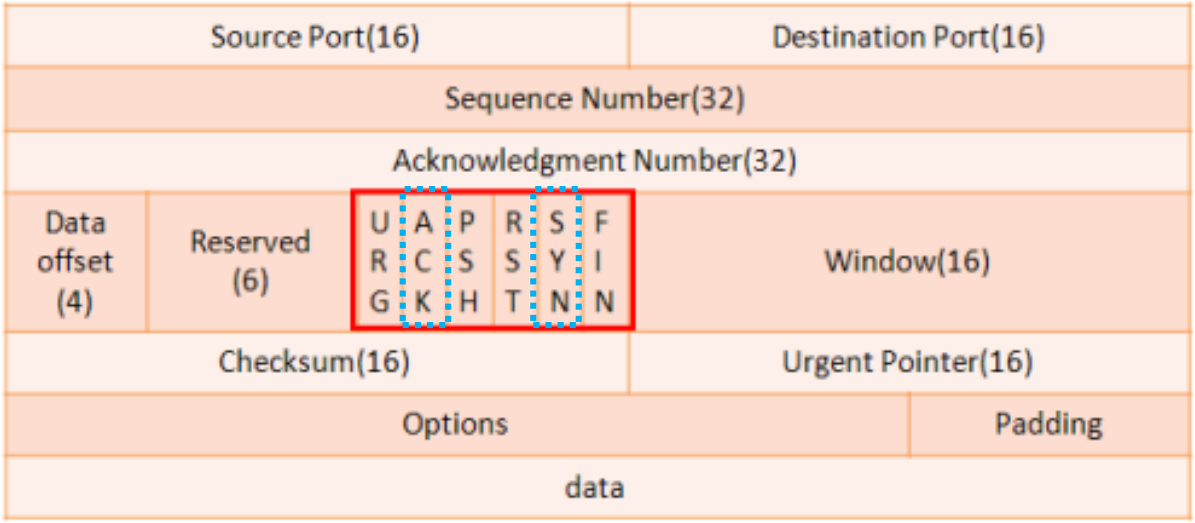


Flags(6 bits)，控制位元旗標

此處最主要有包括以下六個不同功能的位元，只要該位元被設定為1時，表示此封包內有包括該訊息，反之則沒有

例如前面所述的『三向交握』

- ✓ 第一個封包是SYN，則SYN的位元會被設為1，其他被設為0
- ✓ 第二個封包是SYN+ACK，則SYN與ACK的位元會被設為1，其他被設為0
- ✓ 第三個封包是ACK，則ACK的位元會被設為1，其他被設為0



RST(Reset)，重設

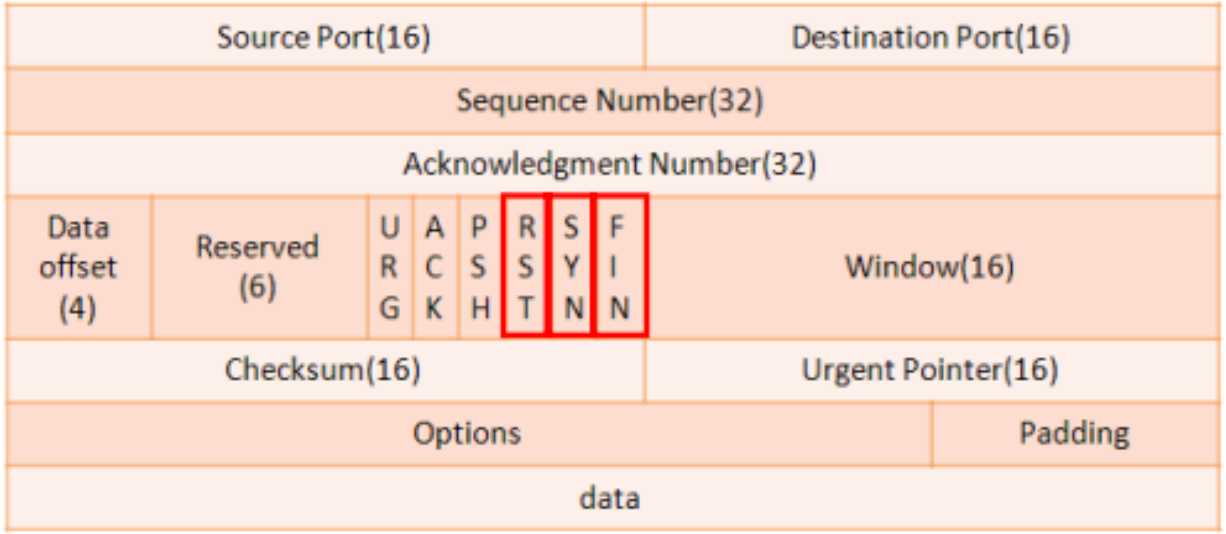
- 通常在彼此雙方發生不預期的錯誤時，會由發生錯誤的一方設定RST為1，告知對方連線已被重設

SYN(Synchronize)，同步

- 當SYN被設為1時，表示這個封包代表是一個同步的訊息

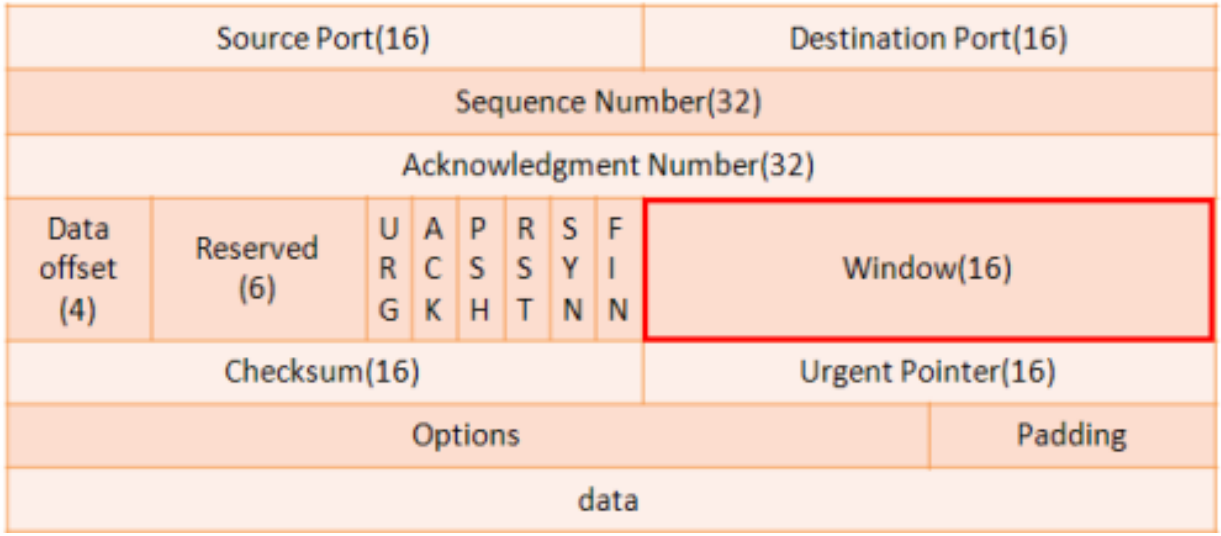
FIN(Finish)，完成

- 當傳送端將資料傳送完畢之後，便會送出完成的訊息給對方，此時就會將FIN設定為1，表示此封包為告知對方已傳送完畢



Window(16 bits)，視窗大小

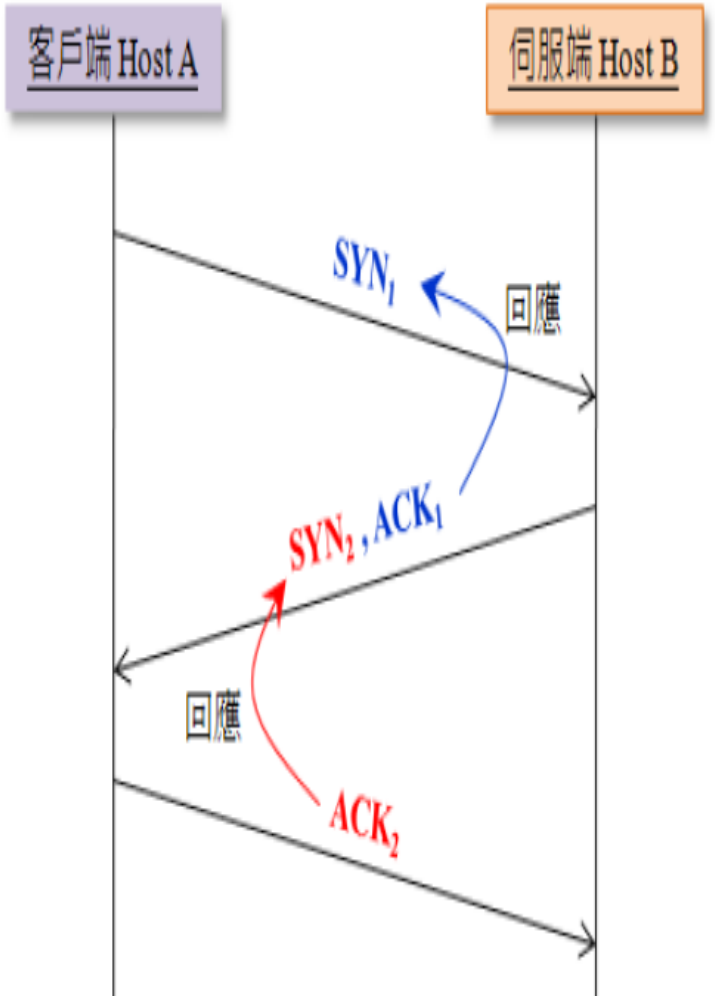
- 長度為16 bits，此欄位代表『接受視窗』(receive window) 的大小，也就是用來告知對方，一次可接受的視窗大小
- 此數值大小的改變可以用來控制對方的流量大小，如果接受端過於忙碌或無法處理進來的大量封包，則會將此『接受視窗』的值調整變小，直到能力足以處理為止，不致於將資料摒棄(discard)
- 反之，則會將此值調整變大，方便對方儘量傳送過來



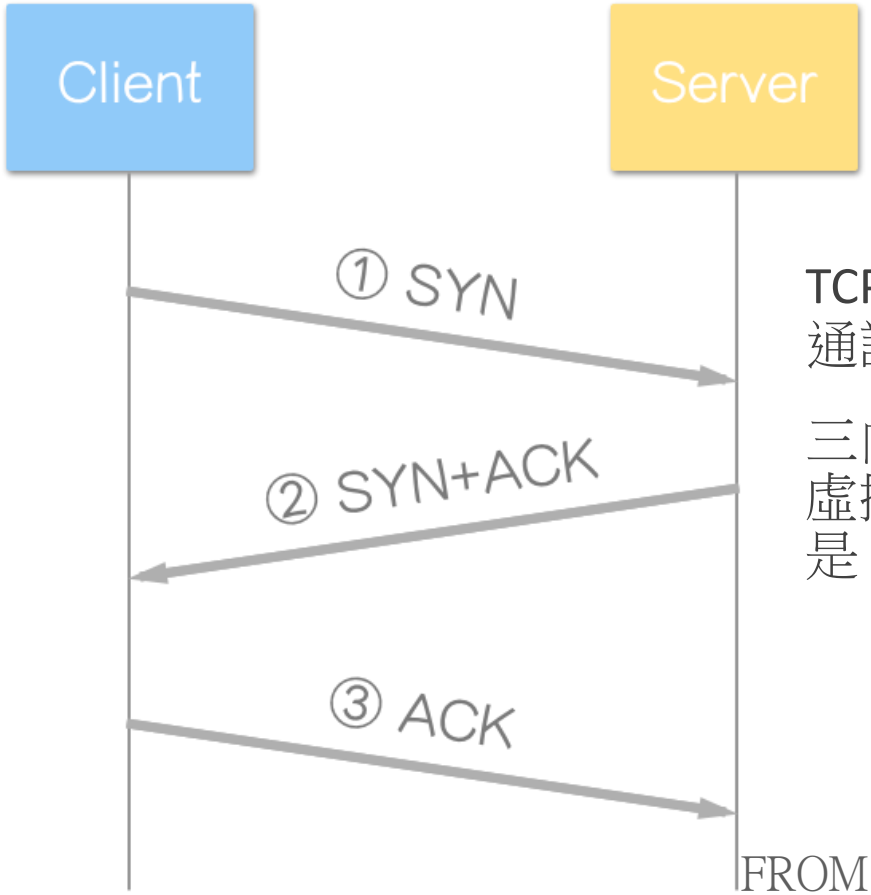
TCP傳輸

傳送單位：區段

連線機制：三方(向)交握



三方交握

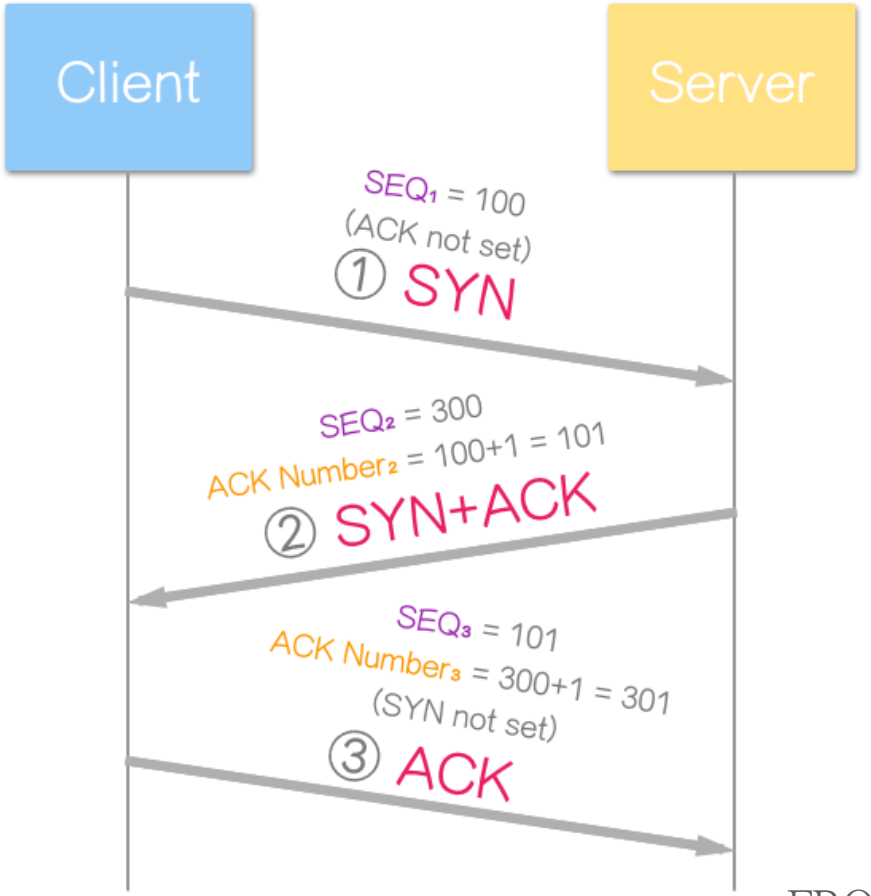


TCP 是一種 連接導向 (connection-oriented) 的通訊協定，

三向交握 (Three-way Handshake)， 是其建立虛擬連線 (virtual connection) 的方式，也就是 三次訊息的交換，確認連線的建立。

FROM 鄭中勝

網路三方交握範例



這次的資料中，第一個位元組 的編號 而下次發送的序號，將會是"此次序號 + 資料長度"

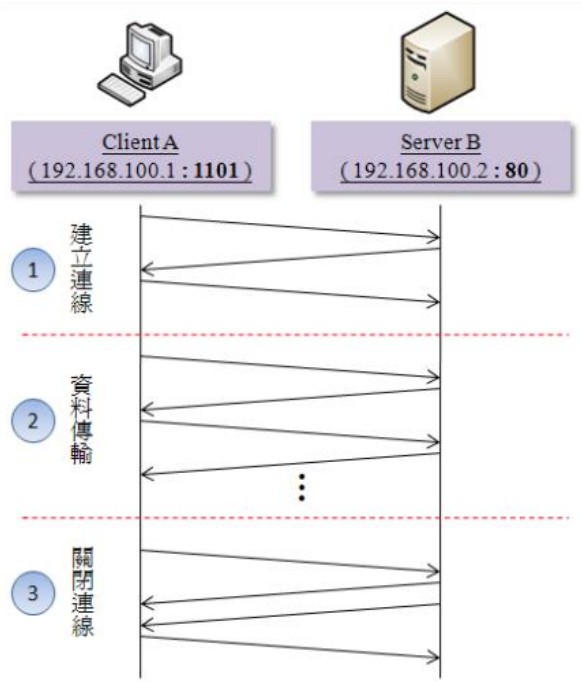
所以對方的確認訊息也將會是"此次序號 + 資料長度"意謂著，

期待下次收到的序號為"此次序號 + 資料長度"

(數值只為範例，不代表實際情形，為避免爭議，使用的是官方的範例數值)

FROM 鄭中勝

由封包看TCP封包的三階段



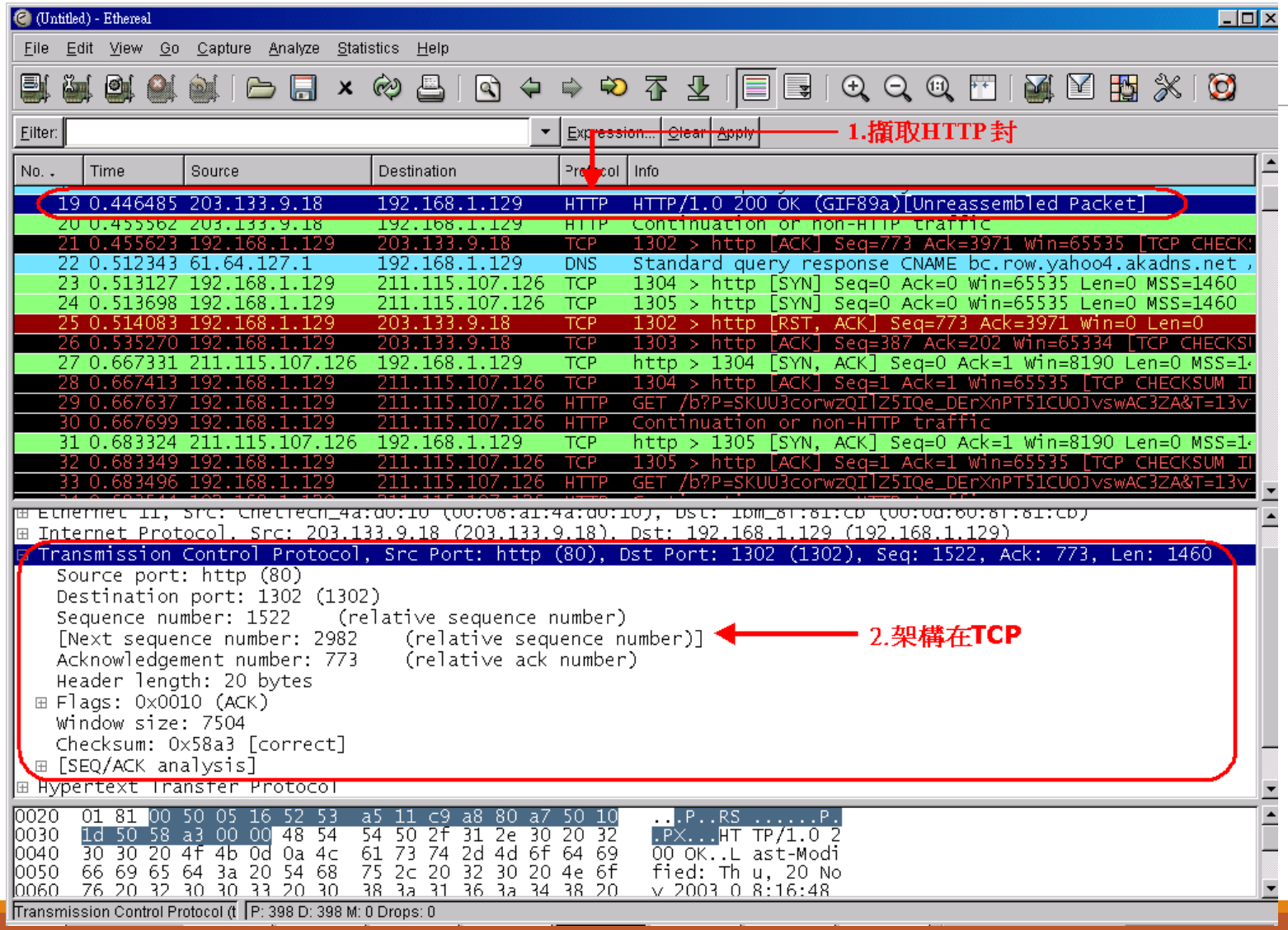
圖中所標示的三個區間，即是TCP的三個階段

- No.3-No.5是"建立連線"階段
- No.6-No.11是"資料傳輸"階段
- No.12-No.14則是"關閉連線"階段

No.	Source	Destination	Protocol	Info
1	00:0c:29:ea:ad:c3	ff:ff:ff:ff:ff:ff	ARP	Who has 192.168.100.2? Tell 192.
2	00:0c:29:53:56:4d	00:0c:29:ea:ad:c3	ARP	192.168.100.2 is at 00:0c:29:53:5
3	192.168.100.1	192.168.100.2	TCP	1101 > 80 [SYN] Seq=0 Win=65535 L
4	192.168.100.2	192.168.100.1	TCP	80 > 1101 [SYN, ACK] Seq=0 Ack=1
5	192.168.100.1	192.168.100.2	TCP	1101 > 80 [ACK] Seq=1 Ack=1 Win=6
6	192.168.100.1	192.168.100.2	HTTP	GET /apache2-default/ HTTP/1.1
7	192.168.100.2	192.168.100.1	TCP	80 > 1101 [ACK] Seq=1 Ack=312 Win
8	192.168.100.2	192.168.100.1	HTTP	HTTP/1.1 304 Not Modified
9	192.168.100.1	192.168.100.2	HTTP	GET /favicon.ico HTTP/1.1
10	192.168.100.2	192.168.100.1	HTTP	HTTP/1.1 404 Not Found (text/htm
11	192.168.100.1	192.168.100.2	TCP	1101 > 80 [ACK] Seq=508 Ack=742 W
12	192.168.100.2	192.168.100.1	TCP	80 > 1101 [FIN, ACK] Seq=742 Ack=
13	192.168.100.1	192.168.100.2	TCP	1101 > 80 [ACK] Seq=508 Ack=743 W
14	192.168.100.1	192.168.100.2	TCP	1101 > 80 [RST, ACK] Seq=508 Ack=

建立連線　資料傳輸　關閉連線

常用的HTTP(網頁)協定就是使用連接導向的TCP封包



通訊埠號

通訊埠號

- 「埠號」(Port) 概念的應用如同港口中的碼頭，由於各碼頭可能同時進行貨物的運送，因此為了正確進行上下 貨動作必須事先確認運作的碼頭為
  - 進行資料傳輸的每台主機所擁有的唯一IP就如同是港口 一般，由於主機可以同時針對不同網路應用(協定)進行 資料傳輸，因此在兩台主機欲互相傳輸資料時同樣須先 確認在哪一個埠號進行處理，方可快速且正確的找到所 對應的傳輸資料
- 也可以把 IP 位址看成主機的門牌號碼，而埠號則是幾樓住戶。

通訊埠號

通訊埠號(Port)

- 在傳輸層協定裡面，為程式產生的程序分配一個通訊埠號(Port)，其值為一個正整數，從0至65535。

網路通訊的兩種模式

- 主動連線 : 主動連線是當埠號建立之後，程序透過該埠號主動發出連線的要求。客戶端使用。
- 被動連線 : 被動模式則是，當埠號建立之後，程序在該埠號等待連線的請求。伺服器端使用。

埠號的編號原則

埠號的編號原則

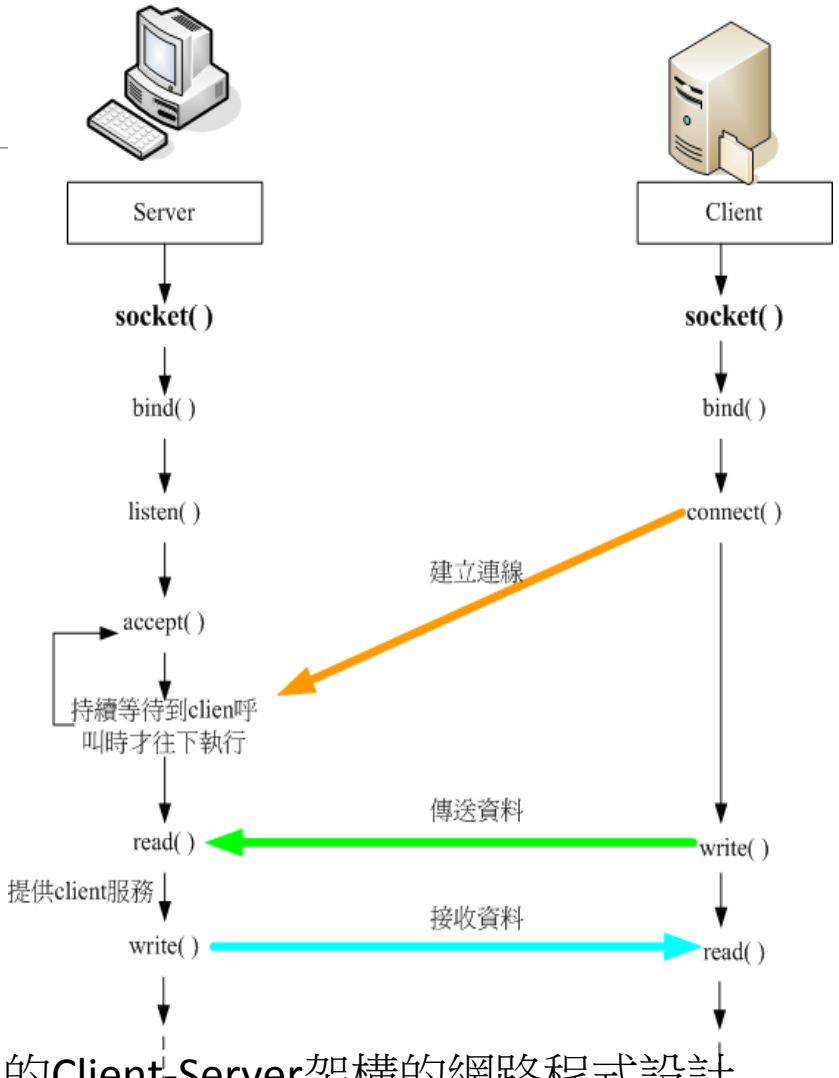
- 常用埠（Well Known Ports）
  - 從0到1023，通常這些通訊埠明確的表明了某種服務的協定。

    例如：80埠實際上總是用來HTTP通訊。
- 註冊埠（Registered Ports）
  - 從1024到49151。此區段留給各軟體公司向IANA註冊。
- 動態或私有埠（Dynamic and/or Private Ports）
  - 從49152到65535。隨機指定給應用程式。

插座對

Socket(插座)

- 根據 IP 來區別主機、根據埠號(port)來區別程式。
- Socket 就是由一個 IP 與一個 Port 所組成的，可將之視為程式與 TCP/IP 連線之間的界面。



常見的Client-Server架構的網路程式設計

常用通訊埠號

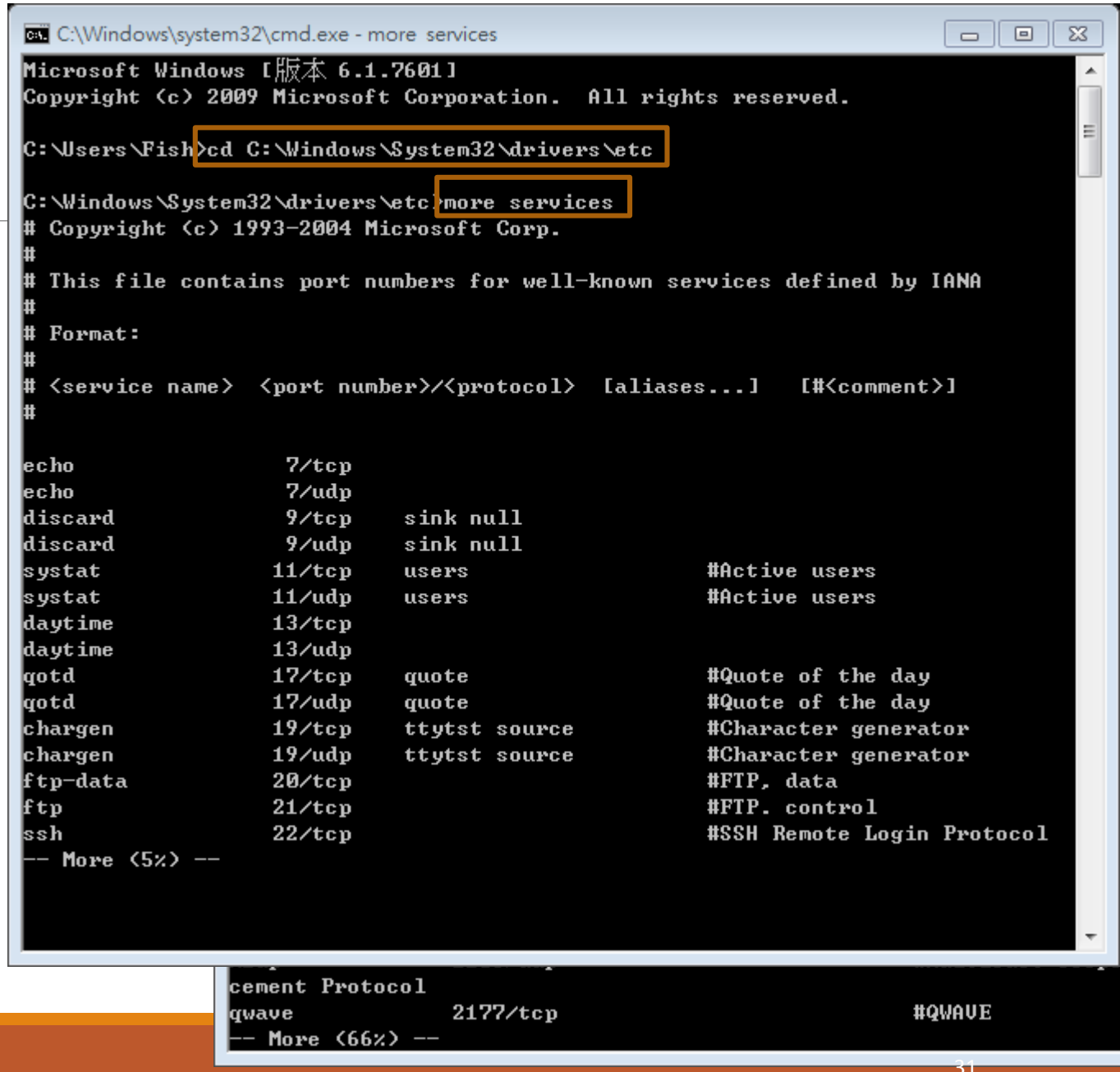
支援服務（應用層協定）	常用埠號	傳輸層協定	說明
FTP DATA	20	TCP	檔案傳輸協定-資料
FTP CONTROL	21	TCP	檔案傳輸協定-控制
TELNET	23	TCP	遠端登入協定
SMTP	25	TCP	EMAIL收信系統
DNS	53	TCP	網路名稱系統
HTTP(WWW)	80	TCP	超文件(網頁)傳輸協定
POP3	110	TCP	EMAIL送信協定
SNMP	161	UDP	簡單網路管理通訊協定
RIP	520	UDP	路由資訊通訊協定

實作：
查詢通訊埠號

協定與port

- 開啟「命令提示字元」
- 將路徑切換到「C:\Windows\System32\drivers\etc」
  - 使用cd + tab
- 查詢ip與port 「more services」

畫面只有出現5%，按「空格」可往下看



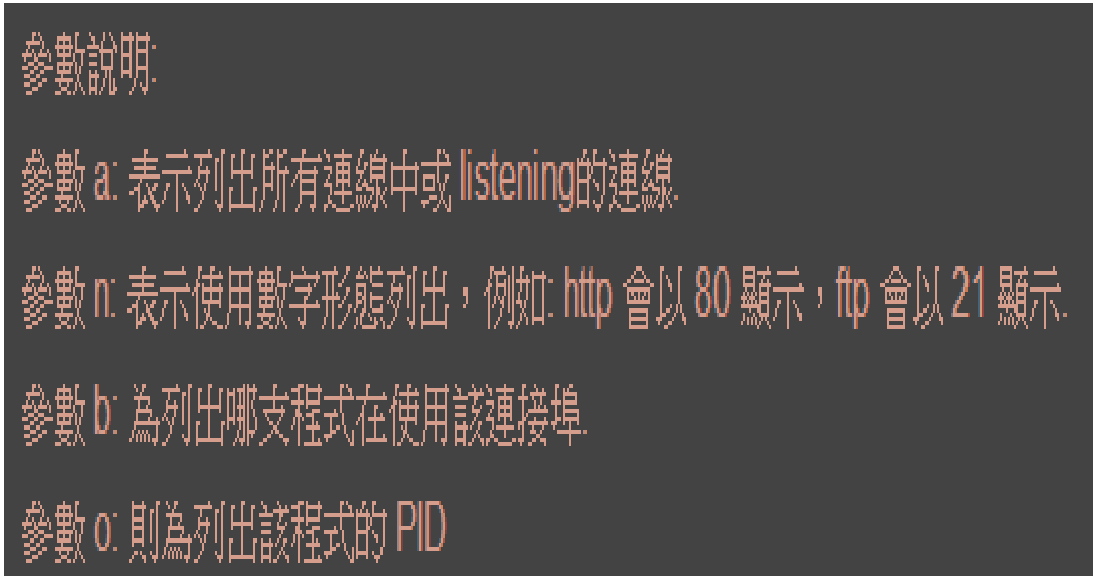
查詢運作的port

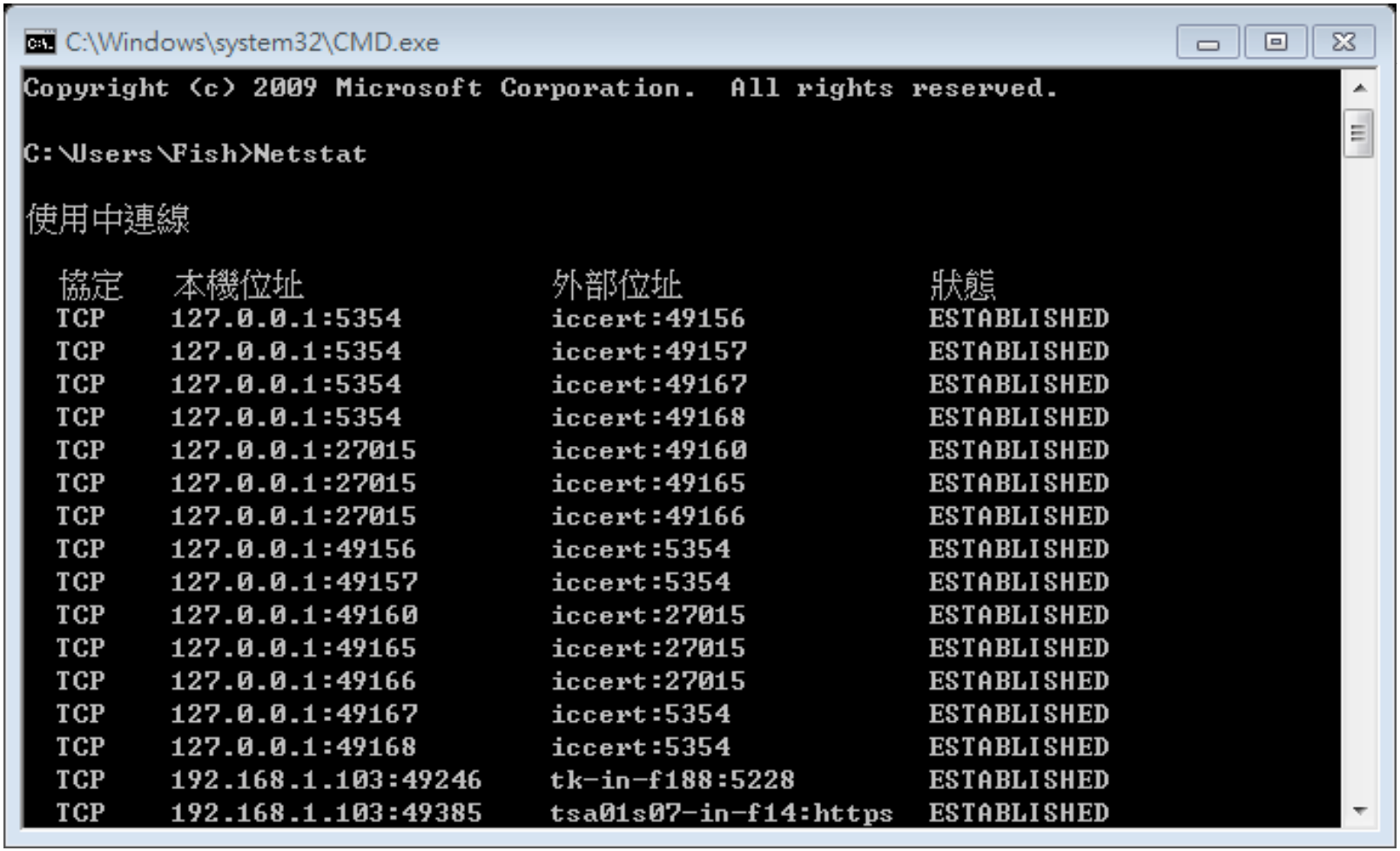
指令：

基本：Netstat

加強： –a/b/n/o

查詢：




C:\Windows\system32\CMD.exe
Copyright (c) 2009 Microsoft Corporation. All rights reserved.
C:\Users\Fish>Netstat
使用中連線
協定 本機位址 外部位址 狀態
TCP 127.0.0.1:5354 iccert:49156 ESTABLISHED
TCP 127.0.0.1:5354 iccert:49157 ESTABLISHED
TCP 127.0.0.1:5354 iccert:49167 ESTABLISHED
TCP 127.0.0.1:5354 iccert:49168 ESTABLISHED
TCP 127.0.0.1:27015 iccert:49160 ESTABLISHED
TCP 127.0.0.1:27015 iccert:49165 ESTABLISHED
TCP 127.0.0.1:27015 iccert:49166 ESTABLISHED
TCP 127.0.0.1:49156 iccert:5354 ESTABLISHED
TCP 127.0.0.1:49157 iccert:5354 ESTABLISHED
TCP 127.0.0.1:49160 iccert:27015 ESTABLISHED
TCP 127.0.0.1:49165 iccert:27015 ESTABLISHED
TCP 127.0.0.1:49166 iccert:27015 ESTABLISHED
TCP 127.0.0.1:49167 iccert:5354 ESTABLISHED
TCP 127.0.0.1:49168 iccert:5354 ESTABLISHED
TCP 192.168.1.103:49246 tk-in-f188:5228 ESTABLISHED
TCP 192.168.1.103:49385 tsa01s07-in-f14:https ESTABLISHED