

2. 網路概念回顧

2.1 網路通訊協定與標準

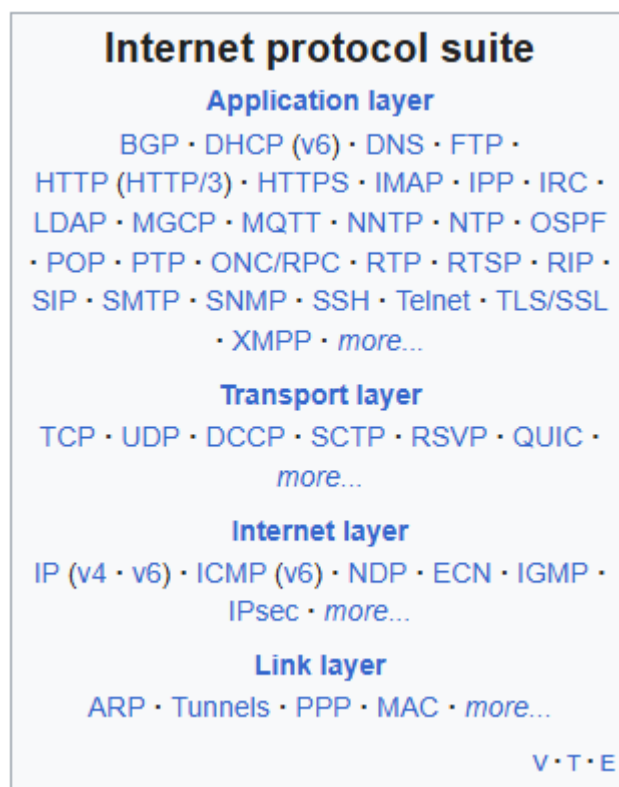
Quick Review: OSI & TCP/IP

快速回憶

- **OSI**: 由國際標準化組織 (ISO) 提出的「概念性分層藍圖」, 用來教學與設計協定
- **TCP/IP**: 由美國國防部國防高等研究計劃署 (DARPA) 主導研發, 實際運作於網際網路的協議套組, 目前全球網路皆以此為基礎

名詞釐清

- 協定 (**protocol**): 一條規則, 規定電腦之間怎麼講話
- 模型: 為了方便管理協定, 設計了一個藍圖, 為這些協定的功能做好分類
- 網際網路協議套組 (**protocol suite**): 一大包能合作的協定集合

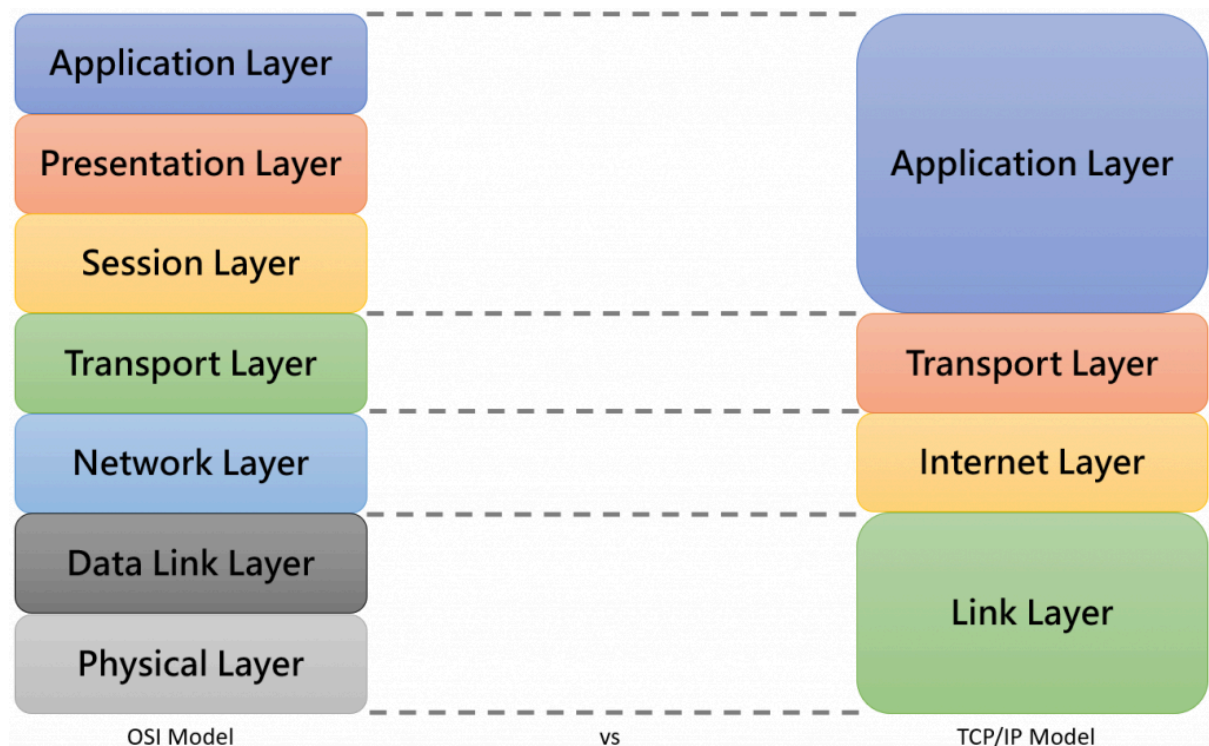


[TCP/IP Internet protocol suite](#)

以 TCP/IP 為例子一句話描述三者關係:

「協定是單一的通訊規則, TCP/IP 模型是分層藍圖, 而 TCP/IP 協定套組則是按照這個架構組合起來、能共同運作的所有協定集合。」

OSI VS TCP/IP



OSI

層級	名稱 (英文)	主要功能
7	應用層 (Application)	提供使用者與應用程式介面
6	表示層 (Presentation)	資料格式轉換、加密、壓縮
5	會議層 (Session)	管理連線與對話、同步控制
4	傳輸層 (Transport)	端對端傳輸、可靠性控制
3	網路層 (Network)	路由與邏輯位址管理
2	資料鏈路層 (Data Link)	錯誤檢測、實體位址控制
1	實體層 (Physical)	傳輸原始位元流、電氣訊號

TCP/IP

第一層: 連接層 (Link Layer)

把電腦的資訊透過網路線或無線「送出去」, 確保資料能到對方電腦

- 封裝資料並處理錯誤檢測
- 控制節點存取與 MAC 地址管理
- 例子/技術:
 - 網路線 (Ethernet cable)
 - 光纖 (Fiber optics)

- 無線傳輸 (Wi-Fi, Bluetooth)
 - Switch
 - ARP (Address Resolution Protocol)
-

第二層: 網際網路層 (Internet Layer)

像衛星導航, 決定資料從哪裡到哪裡走

- 負責資料封包 (Packet) 的路徑選擇 (Routing)
 - 分配邏輯地址 (IP Address)
 - 例子/技術:
 - IP (IPv4 / IPv6)
 - ICMP
 - Router
-

第三層: 傳輸層 (Transport Layer)

像郵差, 確保信件送到對的目的並完整

- 端到端傳輸, 確保資料完整無誤
 - 流量控制、錯誤修正
 - 例子/技術:
 - TCP (可靠傳輸)
 - UDP (不可靠傳輸, 低延遲)
 - 端口號 (Port)
-

第四層: 應用層 (Application Layer)

直接提供使用者看到的功能與服務

- 提供應用程式介面與服務
- 處理資料格式轉換、加密等 (原本 OSI 的表現層/會議層功能大多整合在這裡)
- 例子/技術:
 - HTTP / HTTPS
 - SMTP / POP3 / IMAP
 - DNS
 - SSL / TLS
 - JPEG / MPEG
 - ASCII / Unicode

補充: RFC

網際網路協定的具體規則並不是模型本身制定的, 而是由 **RFC (Request for Comments)** 文件來規範。

- What is RFC:
由網際網路工程任務組 (**Internet Engineering Task Force, IETF**) 發佈的公開技術文件, 用來定義、說明與標準化 網際網路上的各種協定
- OSI、TCP/IP、RFC
OSI 與 TCP/IP 是 "架構藍圖" 的感覺, RFC 則具體規範了協定的細節
- 範例

層級	協定	對應 RFC
網路層	IPv4	RFC 791
傳輸層	TCP	RFC 793
應用層	DNS	RFC 1035

TCP header

2.2 其他基礎概念

IP 位址 (IP Address)

- 網路上每個裝置的 邏輯位址
- 分為:
 - IPv4 (32 位元, 例: 192.168.1.1)
 - IPv6 (128 位元, 例: 2001:db8::1)
- 用來辨別裝置並協助資料封包傳送到正確目的地

IPv4 位址分類與用途

IPv4 位址分類與主要資訊

類別	公共 IP 範圍	私有 IP 範圍	預設遮罩	主要用途
A	1.0.0.0 – 126.255.255.255	10.0.0.0 – 10.255.255.255	/8	大型網路 (早期大型企業、ISP)
B	128.0.0.0 – 191.255.255.255	172.16.0.0 – 172.31.255.255	/16	中型網路 (學校、大型機構)
C	192.0.0.0 – 223.255.255.255	192.168.0.0 – 192.168.255.255	/24	小型網路 (家庭、辦公室)

D	224.0.0.0 – 239.255.255.255	-	-	多播 (Multicast), 一對多傳輸
E	240.0.0.0 – 255.255.255.255	-	-	保留給實驗用途(目前一般網路設備不使用)

IPv4 保留網段

範圍	用途
0.0.0.0/8	廣播資訊到當前主機
10.0.0.0/8	用於私有網路中的本地通訊
127.0.0.0/8	專為主機本身做使用
169.254.0.0/16	DHCP 分配失敗產生的 IP
172.16.0.0/12	用於私有網路中的本地通訊
192.168.0.0/16	用於私有網路中的本地通訊
224.0.0.0/4	用於多播

IPv6 位址與縮寫

- 長度:128 位元, 用十六進位表示, 每 16 位元以 : 分隔
- 範例:2001:0db8:0000:0000:0000:ff00:0042:8329
 - 2001:db8::/32 用於範例文件(documentation)
- 縮寫規則:
 - :: 表示一段連續的零, 但 只能出現一次
 - 範例:2001:db8::ff00:42:8329

特殊位址

前綴	用途
::1/128	回送位址 (Loopback), 對應 IPv4 的 127.0.0.1
fe80::/10	Link-local , 自動分配給每個介面, 用於本地網段通訊, 不可路由到網際網路
2001:db8::/32	文件範例 (Documentation), 不可用於真實網路

- 自動分配：
 - IPv6 Link-local 可透過 **SLAAC** (Stateless Address Autoconfiguration) 自動分配, 不需要 DHCP

過渡技術 (IPv4 ↔ IPv6)

- **4to6** (NAT64 / DNS64) : IPv4 Client 透過轉換訪問 IPv6 Server
- **4in6** (IPv4 封裝成 IPv6) : IPv4 封裝在 IPv6 網路中傳送

子網路 (Subnet)

目的: 減少廣播範圍, 提高效率

- 將一個大型網路切分成多個小網路
- 使用 子網路遮罩 (**Subnet Mask**) 來區分「網路位址」與「主機位址」
- CIDR Value Classless Inter-Domain Routing: 用來表示子網路遮罩長度的數值

常見 CIDR 對照表

CIDR Value	子網路遮罩 (Subnet Mask)	網路位元數	主機位元數	可用主機數
/8	255.0.0.0	8	24	16,777,214
/16	255.255.0.0	16	16	65,534
/24	255.255.255.0	24	8	254
/25	255.255.255.128	25	7	126
/26	255.255.255.192	26	6	62
/30	255.255.255.252	30	2	2

可用主機數 = 2 的 (32-CIDR) 次方 - 2
 減 2 代表減去網路位址與廣播位址

情境範例: 子網路為 172.16.0.0/16

類型	IP 位址	說明
網路位址	172.16.0.0	代表整個 /16 子網路, 不能分配給主機
第一個可用主機 IP	172.16.0.1	可以分配給主機
最後一個可用主機 IP	172.16.255.254	可以分配給主機

廣播位址 172.16.255. 用於對整個 /16 子網內廣播, 不能分配給主機
255

Routing Table 與選徑

- 路由器根據 **Routing Table** 決定下一跳
- **Longest Prefix Match**(最長前綴匹配): 選擇與目的 IP 最長相符的路由
- 若多條路由長度相同, 依據 **weight / metric / cost** 決定
- 路由選擇(前綴長度相同時)

依據	說明	數值大小與優先
Weight	廠商自訂值(如 Cisco), 用來決定同前綴路由優先順序	數值越大 → 優先級越高
Metric / Cost	路由成本或距離, 通常依 Hop Count (跳數)、延遲或頻寬計算	數值越小 → 優先級越高

- 封包選徑可能非對稱: 去與回的路徑不必相同
-

ICMP(Internet Control Message Protocol)

- 功能: 傳遞控制訊息與回報網路層錯誤, 協助診斷與排錯
 - 常用工具:
 - **ping**: 測試主機是否可達
 - **traceroute**: 追蹤封包路徑
-

traceroute 運作原理

1. 送出一系列封包(通常為 UDP、ICMP 或 TCP), 並逐步增加封包的 **TTL (Time-to-Live)**
 2. 封包到達路由器時, TTL 減 1; 若減到 0, 該路由器會回送 **ICMP Time Exceeded (Type 11, Code 0)**
 3. 根據回覆中的來源 IP, 即可得知該「跳(hop)」的路由器
 4. 程式持續增加 TTL(例如 1 → 2 → 3...), 直到抵達最終目的地
-

常見 ICMP Type

Type	名稱	用途
------	----	----

0	Echo Reply	ping 的回覆
3	Destination Unreachable	無法到達目的地(含多種 Code, 例如 Port Unreachable)
8	Echo Request	ping 的請求
11	Time Exceeded	TTL 歸零, 用於 traceroute

TCP vs UDP

特性	TCP	UDP
可靠性	有確認機制(ACK)、重傳、順序保證	丟過去就不管了(只負責丟)
連線模式	連線導向(需建立連線)	無連線(直接傳送)
速度	較慢(需花費額外的控制成本)	較快(成本低)
適用情境	網頁、電子郵件、檔案傳輸	即時影音串流、遊戲、VoIP

TCP 三向交握(Three-Way Handshake)

1. Client → Server: **SYN**
2. Server → Client: **SYN-ACK**
3. Client → Server: **ACK**

→ 建立連線後開始資料傳輸

→ 關閉連線則為四次揮手(FIN / ACK)

→ 想要主動關閉連線可以直接發送帶有 RST flag 的封包

Packet Size / MSS / MTU

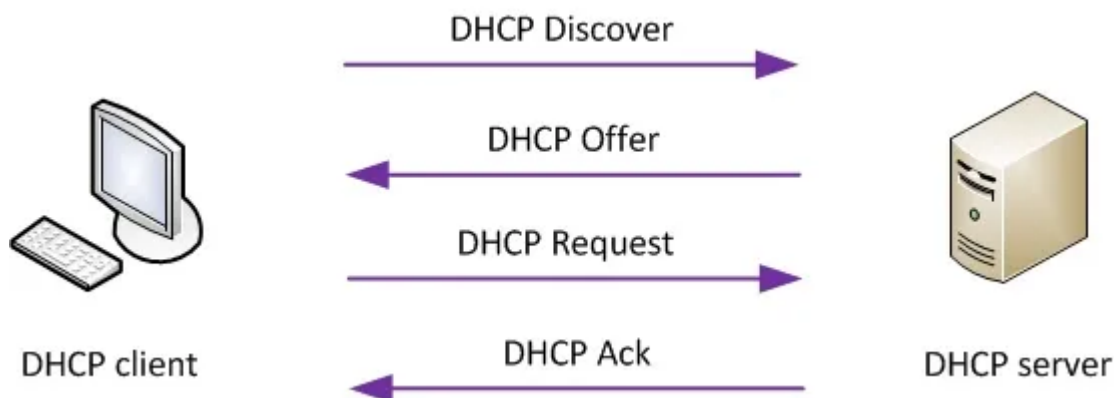
- **MTU(Maximum Transmission Unit)**
 - 網路層中, 單個封包(Frame)可攜帶的最大總長度
 - 以太網路常見 MTU = **1500 bytes**
- **MSS(Maximum Segment Size)**
 - TCP 傳輸層中, 單個 TCP 封包能裝載的「純資料」大小
 - 計算方式: **MSS = MTU - IP Header - TCP Header**
 - 在 IPv4 + TCP 預設下:
 - IP Header: 20 bytes
 - TCP Header: 20 bytes

■ $\Rightarrow \text{MSS} = 1500 - 20 - 20 = 1460 \text{ bytes}$

- 封包過大 \rightarrow 分片(Fragmentation)
 - 若 IP 封包大於路徑上任一鏈路的 MTU, 需切割成多個片段傳送
 - 增加處理負擔與延遲, 效率較差
 - Jumbo Frame(巨幀)
 - 可支援更大的 MTU(常見 9000 bytes)
 - 減少分片次數, 提升大流量傳輸效率(如資料中心、儲存網路)
 - 需要整條鏈路的設備都支援才能使用
-

DHCP(Dynamic Host Configuration Protocol)

- 自動分配:
 - IP 位址
 - 子網路遮罩
 - 預設閘道(Gateway)
 - DNS 伺服器位址
- 讓裝置連上網路時免手動設定網路參數



NAT(Network Address Translation)

- 一種 網路位址轉換技術 將私有 IP 位址轉換成公有 IP 位址
- 常見於路由器上, 讓多個內部裝置共享同一個公網 IP

NAT 解決了 IPv4 公網 IP 位址不足的問題, 也是家用網路連網的常見方式。

DNS(Domain Name System)

- 將網域名稱轉換為 IP 位址
- 例: 輸入 `www.example.com` \rightarrow 解析成 `93.184.216.34`

IP 太多不好記, 需要這個

3. 網卡設定與 IP/Gateway 調整

3.1 設定 IP

先 cd 進 /etc/netplan/

找到前綴數字最大的檔案 cat 出來

- DHCP 自動取得 IP

```
network:
  version: 2
  ethernets:
    enp0s3:
      dhcp4: true
```

- 靜態 IP

1. 編輯 netplan 設定檔:

```
network:
  version: 2
  ethernets:
    enp0s3:
      addresses: [10.0.2.3/24] # 可以試試看原本能連線的那組
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
      dhcp4: no
      routes:
        - to: 0.0.0.0/0
          via: 10.0.2.2
```

-

下 `sudo netplan apply`

如果跳出警告那有可能是 netplan 要求設定檔不能被其他使用者修改，權限太寬鬆會出警告
下指令確保只有 root 可以寫入，其他人可以讀取即可

```
sudo chmod 600 /etc/netplan/檔案
```

```
sudo chown root:root /etc/netplan/檔案
```

接著可以下 `ip a` 觀察 ip 是否有符合設定