第一章IPv6 簡介

Not every flower can represent love, but roses did it
Not every tree can stand thirst, but cactus did it
Not every protocol can fit the future, but IPv6 did it.

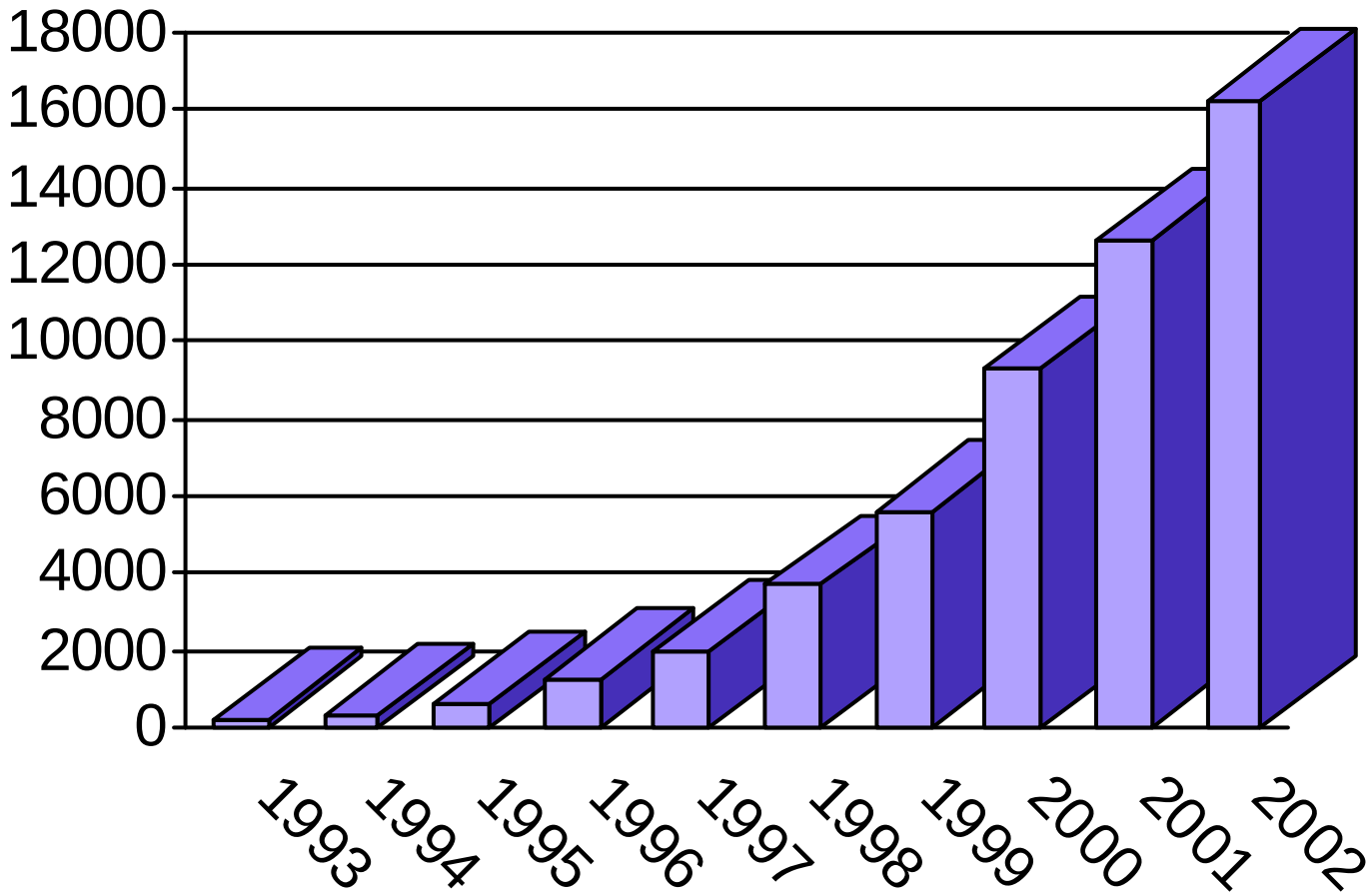
IPv6, the standard of the Internet Future.

章節目錄

網路的趨勢與發展

- Moore’s Law：平均每18個月晶片的容量會成長一倍，而成本卻減少一倍
- New Moore’s Law ─光纖定律：網際網路頻寬每九個月就會增加一倍的容量、而成本降低一半

全球DNS有註冊的主機數量



Internet起源

- 1969 年，美國國防部(DoD) 開立的美國國防部高等研究計劃局(ARPA)建立ARPANet
- 1973 年，全錄公司帕洛亞托研究中心(PARC)，成功發展出乙太網路（Ethernet）
- 1986 年，美國國家科學基金會根據原先 ARPANet 使用的 TCP/IP通訊協定，建立了 NSFNet
- 1991 年，國家科學基金會正式宣佈開放於商業用途使用
- 1993 年， WWW介面問世

台灣的網際網路發展

- 1986年，幾所國立大學院校之相同廠牌的電腦主機加以電信局的數據線路連接，形成 IFNET與UNINET兩個網路。
- 教育部電算中心與台灣IBM公司「教學研究資訊服務」合作計畫，連線到台大等十五個學校，發展至後來建立的TANet
- 1987年，連線日本BITNET國際學術網路，透過與日本東京理科大學（SUT）與全世界BITNET節點互傳訊息

台灣的網際網路發展

- 1989年，教育部電算中心研擬「大學高速學術網路」（TANet)取代BITNET，以FDDI網路為骨幹架構，各校內以Ethernet形成校園區域網路，架構於各主要大學之間。
- 1990年，校際之間網路通訊協定採TCP/IP標準，將此命名為台灣學術網路。以Internet架構之9.6kbps專線串接校際
- 1991年，TANet T1骨幹開始運作，12月以64kbps接通美國美國普林斯頓大學JvNCnet，並可直接連通美國國家科學基金會網路（NSFNET）骨幹，正式開啟了我國Internet的時代。

IPv6 的發展

- 1992年，IETF之IPv4的Address空間不足的問題開始被檢討
- 1994年，下一代的網際網路協定開始被提案，CATNIP (Common Architecture for the Internet)，TUBA (TCP/IP with Bigger Addresses)，SIPP (Simple Internet Protocol Plus)三個提案中出線。
- 1995年，SIPP被更名為IPv6，IPv6的規範將被RFC1752(The Recommendation for the IP Next Generation Protocol)公開。

IPv6 的發展

- 1998年，IPv6之位址架構與通訊協定之規範分別在RFC2373 (IP Version 6 Addressing Architecture)與RFC2460(Internet Protocol Version 6(IPv6) Specification)公開。
- 1999年，全球第一個業界團體(共有42個單位加盟)成立了「IPv6 Forum」。ARIN 將全球第一個之IPv6 Prefix：2001:400::/35授予給ESnet。
- 2002年，全球各區域性的Internet Registry RIR(Regional Internet Registries)實施新的「IPv6 Address Allocation and Assignment Global Policy」。

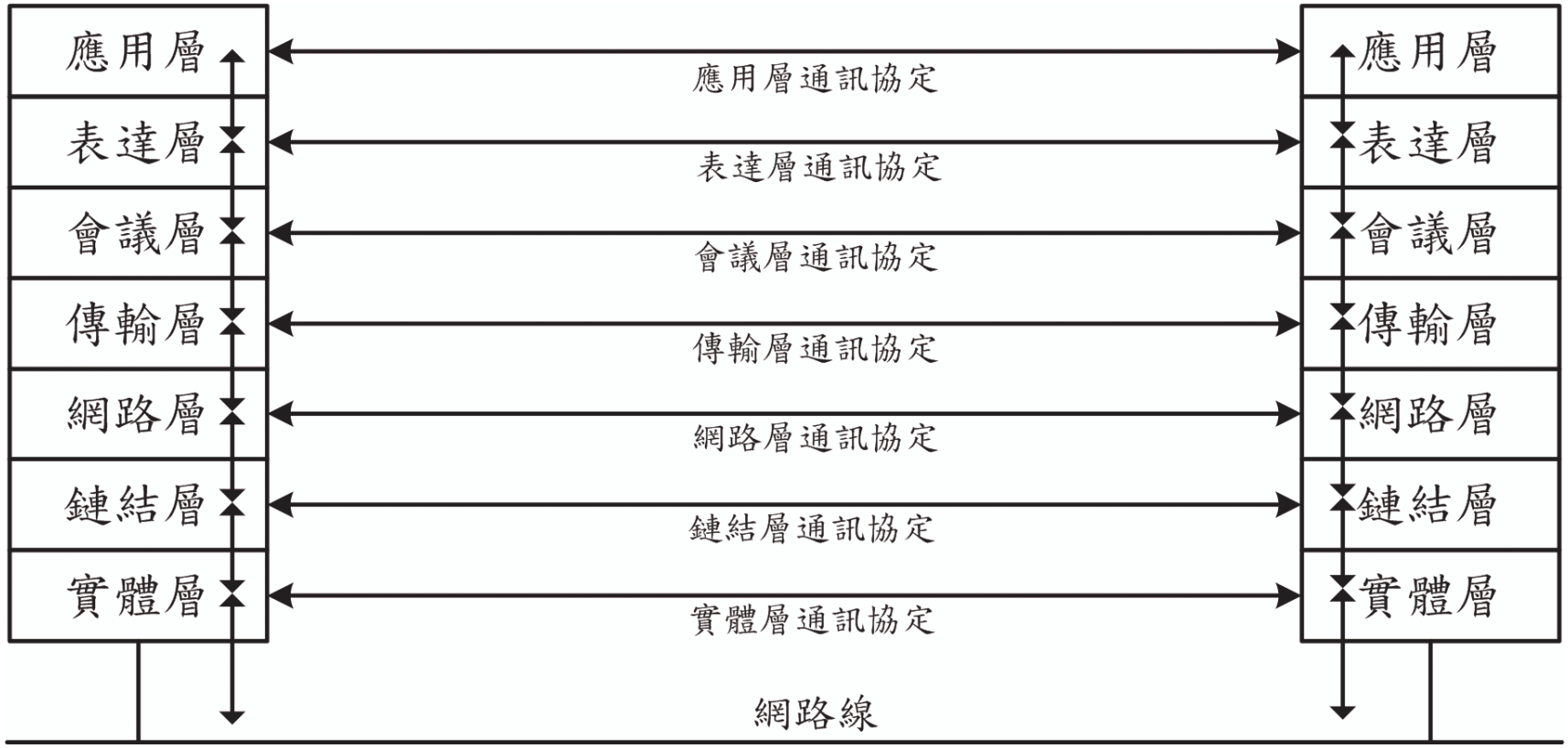
網路的基本協定：OSI與TCP/IP

網際網路的目的是讓電腦能互相溝通。正如人與人相互對話一樣，只懂中文的人是無法跟只懂英文的人對談的，因此，要達成此目的有一必要條件：讓電腦說共同的語言。這樣的語言我們稱之為電腦的通訊協定(Protocol)。

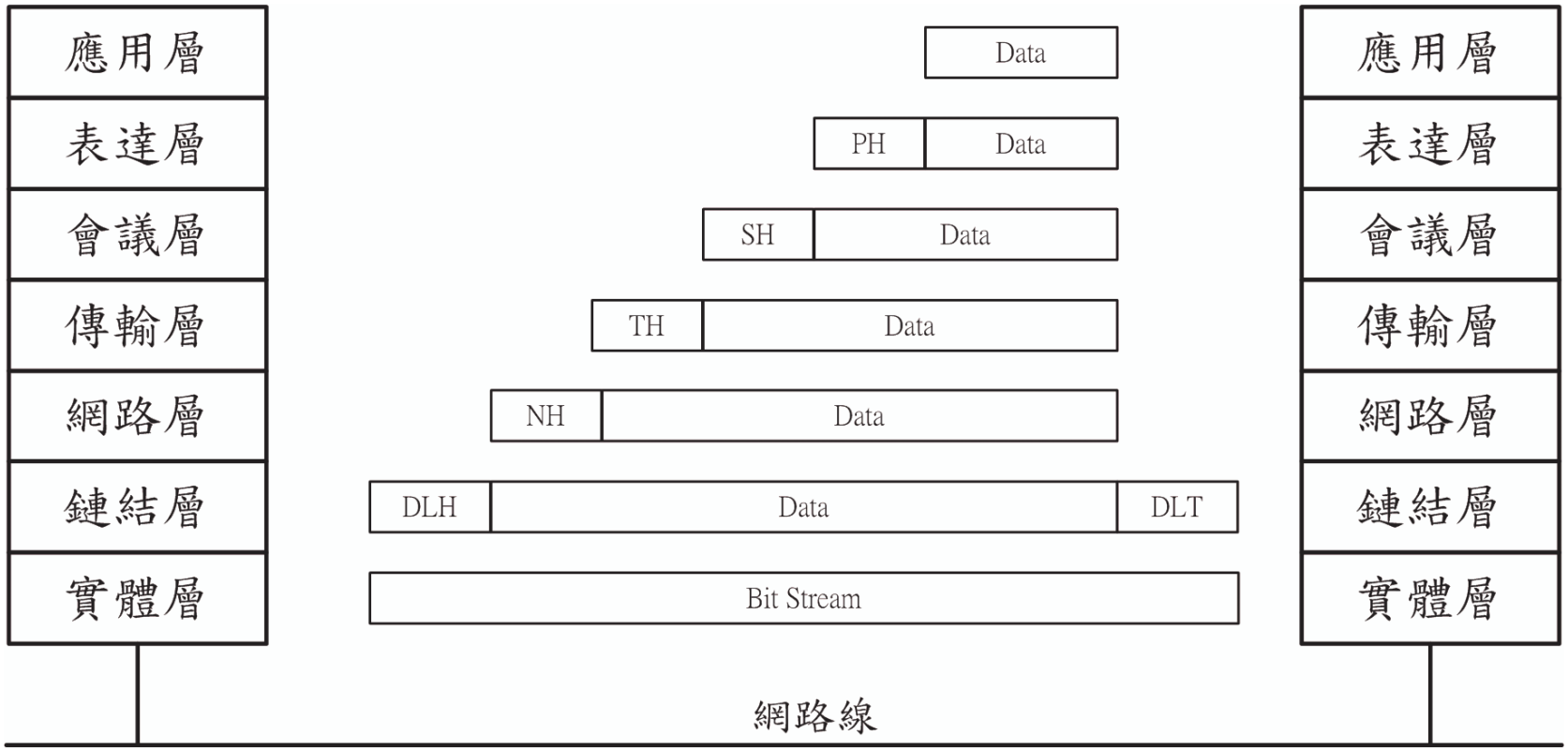
OSI模型

- 1980年國際標準組織ISO(International Organization for Standardization)制定了 OSI (Open System Interconnection)的基本模型。
- 網路架構模型分成7層 (7-Layer Reference Model)
- 這種堆疊式的多層模型即稱作協定堆疊(protocol stack)。

OSI 協定堆疊


應用層
表達層
會議層
傳輸層
網路層
鏈結層
實體層
應用層通訊協定
表達層通訊協定
會議層通訊協定
傳輸層通訊協定
網路層通訊協定
鏈結層通訊協定
實體層通訊協定
網路線
應用層
表達層
會議層
傳輸層
網路層
鏈結層
實體層

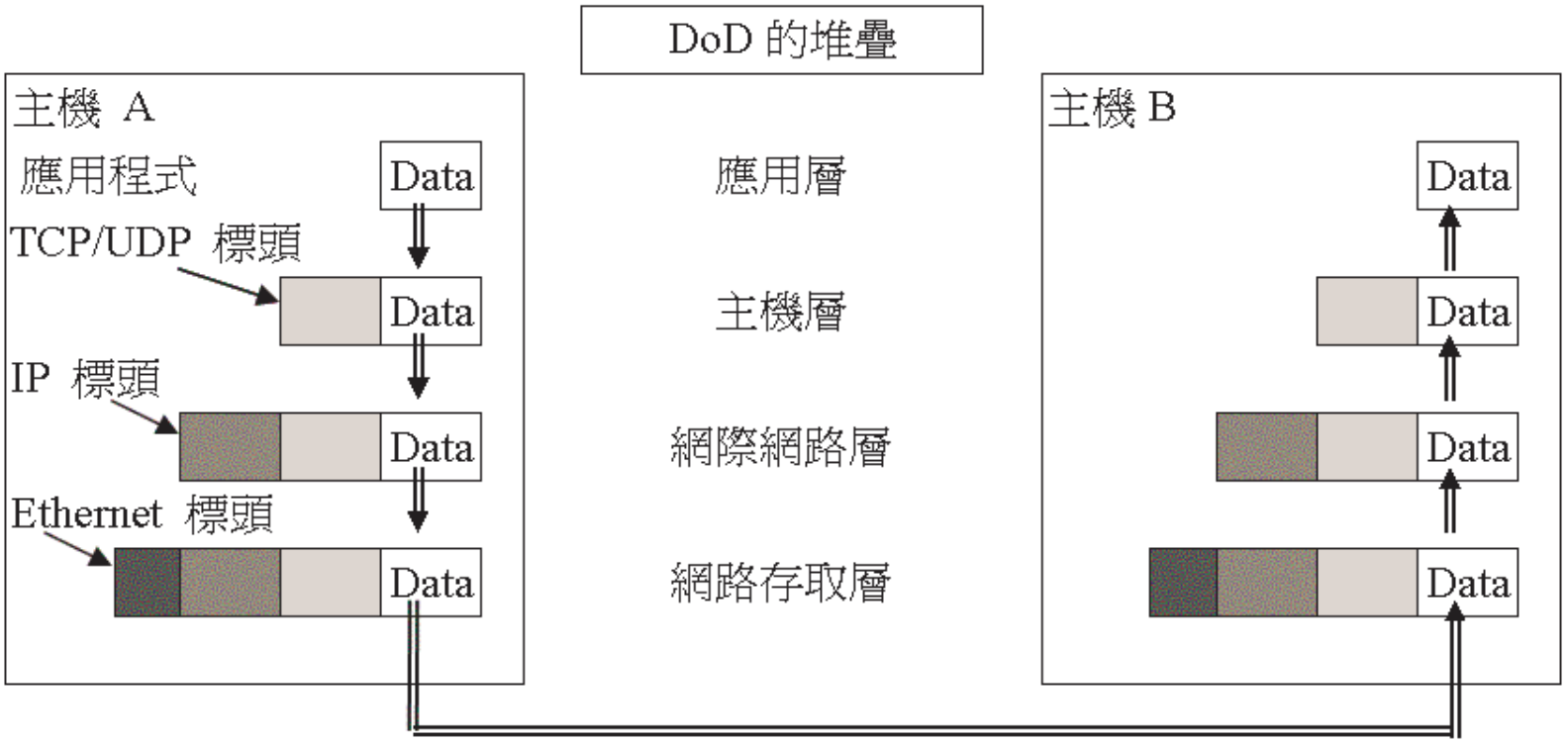
OSI模型層與層間的關係



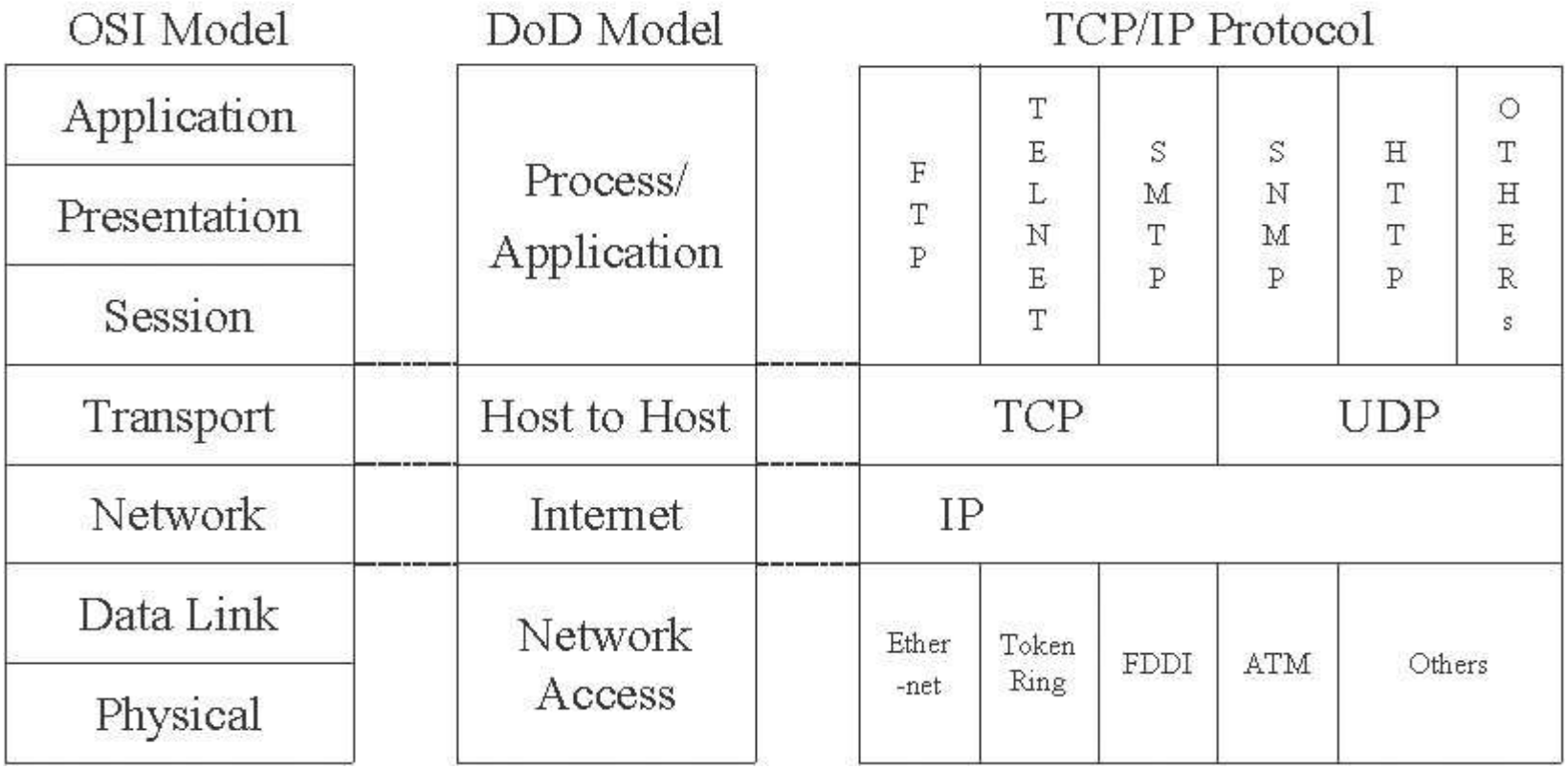
DoD模型

- 當初為了ARPANet實驗計畫開發出來的模型，又稱之為ARPANet模型
- TCP/IP是依此建立出來的，又有人稱之他為TCP/IP模型
- 先成為公認準則後，才正式成為標準文件(RFC: Request For Comments)的一部分，因此也被稱為通信協定的The Fact Standard

DoD模型層與層間的關係



兩種模型與實際協定對照圖



IP 的角色

網路的門牌號碼：IP，位於網路堆疊的中心位置，兼容不同的網路介面，對Transport Protocol或Application提供統一的通訊方式。

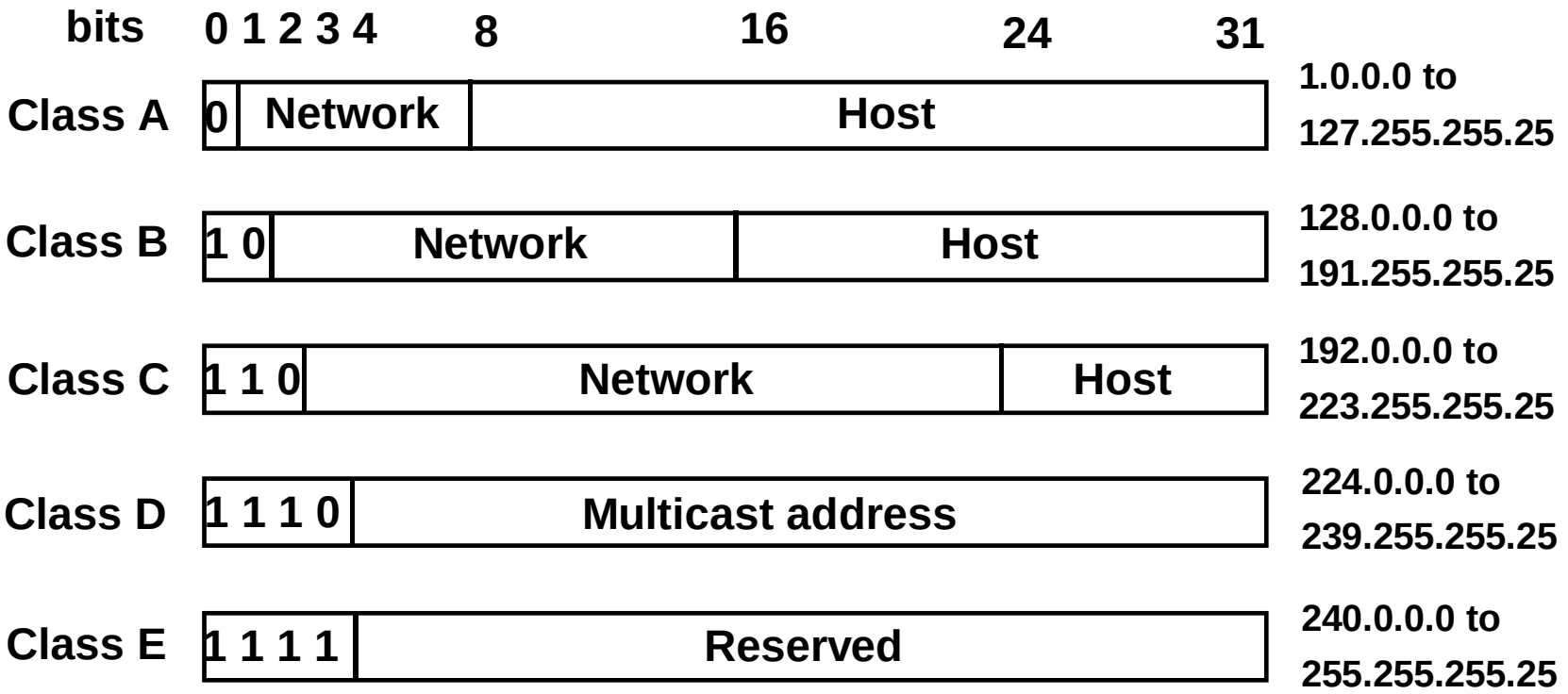
IP 位址分配的組織

- 以紐約的IANA為中心，其下再依區域分成四個區域註冊中心(Regional Internet Registries)，
  - 歐洲地區：RIPE NCC(Réseaux IP Européens Network Coordination Centre)
  - 北美地區： ARIN(American Registry for Internet Numbers)
  - 亞太地區： APNIC(Asia Pacific Network Information Centre)
  - 拉丁美洲：LACNIC(Latin American and Caribbean IP address Regional Registry)

IPv4 位址的分配類別

類別	網路位址	主機位址	最多主機數量	可分配的組織數
A	8位元	24位元	16,777, 214	128
B	16位元	16位元	65,534	16,384
C	24位元	8位元	254	2,097,152

Class-ful IPv4 Address



IP 路由

- 決定要經過何種路徑能到達IP位址所表示的目的地的架構
- Datagram（per packet）
- Hop by Hop
- 參照路徑表

IP 的演進

1981年9月，RFC791制定了Version 4（IPv4）的版本，從誕生迄今大約經過了20年的時間。在這段期間電腦的技術有很大的進步，出現了各式各樣在當初設計IPv4時所未被設想到的使用型態。網路的急速普及化，也使得現在的網路規模遠超過設計當初的預期。

IPv4的20年

- 1981年9月，RFC791規定 IPv4
  - 未被設想到的使用型態－可移動的節點
  - 網路的急速普及化－網路規模遠超過設計當初的預期
- 1990年代前期，IETF開始檢討以下的3個問題
  - 今後網路發展的預測及仰賴IPv4所能維持的極限
  - IPv4的延續策略
  - 設計新的IP及策定轉移計畫

IPv4的極限

- IP位址數的不足
  - 類別基礎的IP位址分配方法，類別C的IP位址是以最後8位元為網路大小，真正單位所需要的IP位址數跟實際上所分配到的IP位址數會有所差距，造成位址分配較沒有彈性。
- 路由表太大
  - 接續上的網路數量增加的話，路由表也會跟著變大
- 解決的方法
  - CIDR
  - NAT
  - IPv6

CIDR

- 廢止IP 分類方式，讓一單位可被分配幾個連續的類別C位址。(subnet大小不一定的8的倍數，而是可變動的)
- 加入維持路徑表大小的技術，利用路由彙整方式來維持路由表的大小
- 遭遇困難：
  - 匯集的網路都必須是集中在附近的。各地域、國家、沿著Network的接續構造來分割區塊進而作分配。這跟目前實際依組織分配，散佈全球的IP 規劃是大不相同的，也使得實際達成的可能性微乎其微

NAT

- Network Address Translation
- 只對出入閘道分配真實位址，內部的主機則採用私有位址的方式
- 以閘道器來做轉換
- 遭遇問題：
  - 沒有真實位址，無法達成P2P通信
  - 安全性問題
  - 效能不彰

朝向IPv6

- IPv4的各式延續策略，讓 IP 的短缺暫時脫離了窘境，然而，這些方法終究會到達極限。
- 新一代的IP協定的考量
  - 相符於 IPv4 於目前協定堆疊中的流程
    - 基本的動作相同
    - 更單純化的Protocol
  - 解決至目前為止的問題點
    - 位址空間的不足
    - Multicast、Mobile
  - 更容易運用
    - Plug and Play
    - Security
  - 可以因應後續長時間的發展
    - 容易擴充新功能
    - 容易自IPv4轉移

IPv6 的發展

- 最初被稱為下一代IP(IPng: IP The Next Generation)的協定
- CATNIP、TUBA與SIPP三種協定，是最被任認可的版本
- IETF最後決定採用SIPP
- 1995年SIPP被更名為IPv6

IPv5？

- Internet的版本序號是由IANA管理
- 第五版被RFC1190(Experimental Internet Stream Protocol, Version 2(ST-Ⅱ))歸為實驗型行的Protocol
- 新的IP 版本就成了第六版
- 版本序號：7、8、9及15都已被預約了，沒被預約的只有10~14。因此當下一次再有IP的下一版本要提出時，應該是用第十版了

IPv6的特徵

- 提供新的定址方式
- 可擴充新的通信協定

位址空間的擴充

版　本	位元數	位　　址　　數　　量
IPv4	32	4,294,967,296個
IPv6	128	340,282,366,920,938,463,463,374,607,431,768,211,456個 (≒$3.4x10^{38}$)

Why not > 128 bits?

- 考慮到IP 標頭處理所造成的浪費。

協　　定	標頭長度	MTU	標頭浪費
IPv4	20bytes	576bytes	3.5%
IPv6	40bytes	1,280bytes	3.1%

- 位址空間擴充了，但標頭浪費卻幾乎不變，可以判斷這樣的位址長為128bits是妥當的

標頭的簡化

- 刪除了許多IPv4的欄位
  - 標頭長度
  - 識別子(Identifier)
  - 分段位移(Fragmentation Offset)
  - 檢查碼(Checksum)
  - 服務類別(Type of Service)
- 減輕網路中路由器的負擔
  - IPv6的基本標頭從可變長度變更成固定長度
  - 取消路由器對封包的分割處理
  - 刪除Checksum 機制

可擴充協定標準

- IPv4使用Option 欄位，但只限於這一個標頭
- IPv6除基本標頭外，可再加上一或多個延伸標頭來形成
- IPv6 Basic 標頭 + Extension 標頭(s) + Data

其他特徵

- 即插即用
  - DHCPv6 - Statefull自動設定
  - Stateless自動設定
- 可並存於原有IPv4的環境下進行通信
- 強化的安全性
  - 內建 IPsec
- 更好的品質管控(QoS)

IPv6標準化狀況

IPv6新世代網際網路協定暨整合技術

IETF 公開的標準

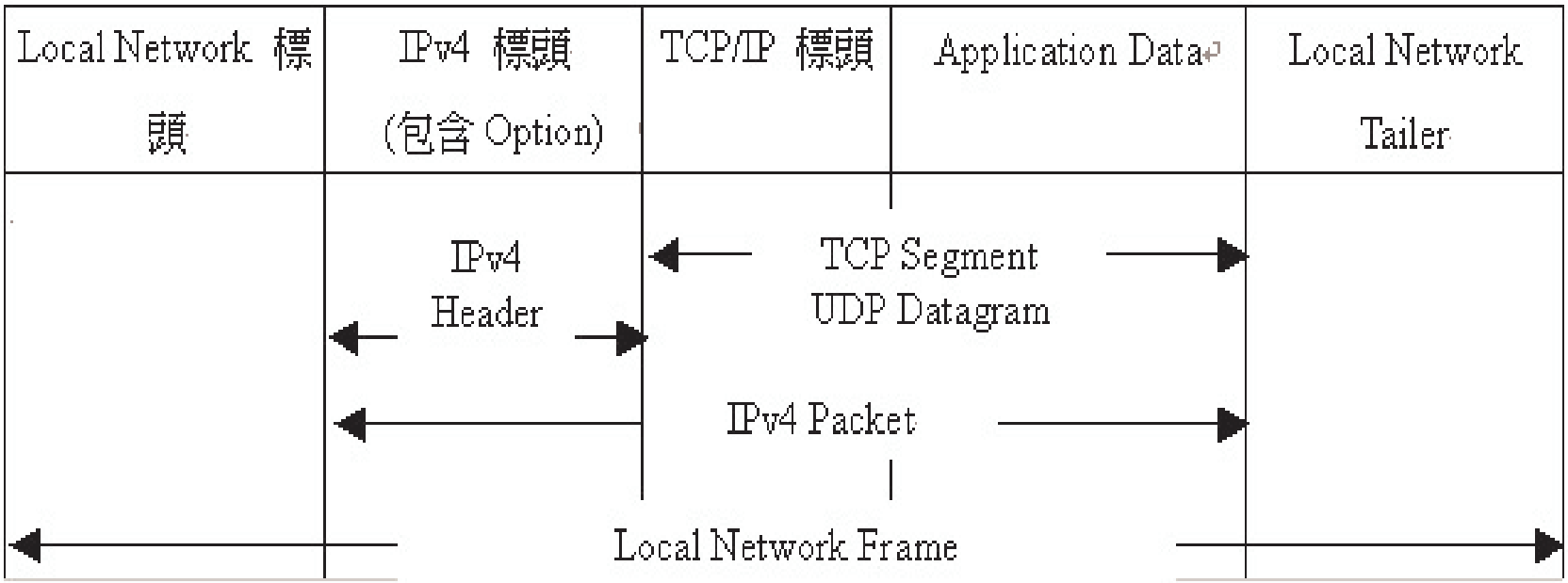
- 首先提案：1995年12月
  - RFC1883(Internet Protocol, Version6(IPv6) Specification)
- 目前的標準規範：1998年12月
  - RFC2460(Internet Protocol, Version6(IPv6) Specification)
- 位址的分配方式： 2003 年4月
  - RFC3513(Internet Protocol Version 6 (IPv6) Addressing Architecture)

IPv6的基本架構

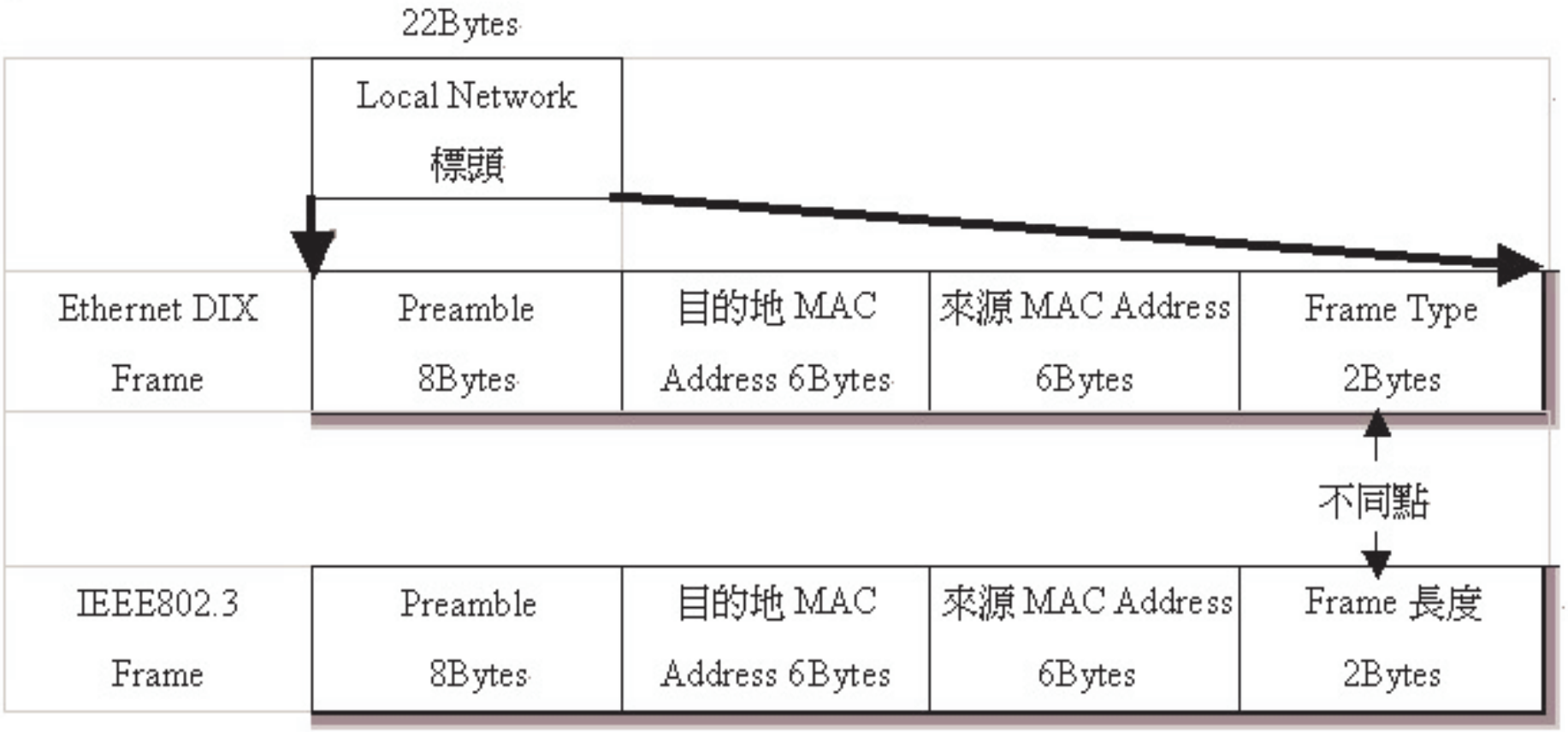
IPv4 vs. IPv6

IPv6新世代網際網路協定暨整合技術

IPv4 Packet 傳送架構



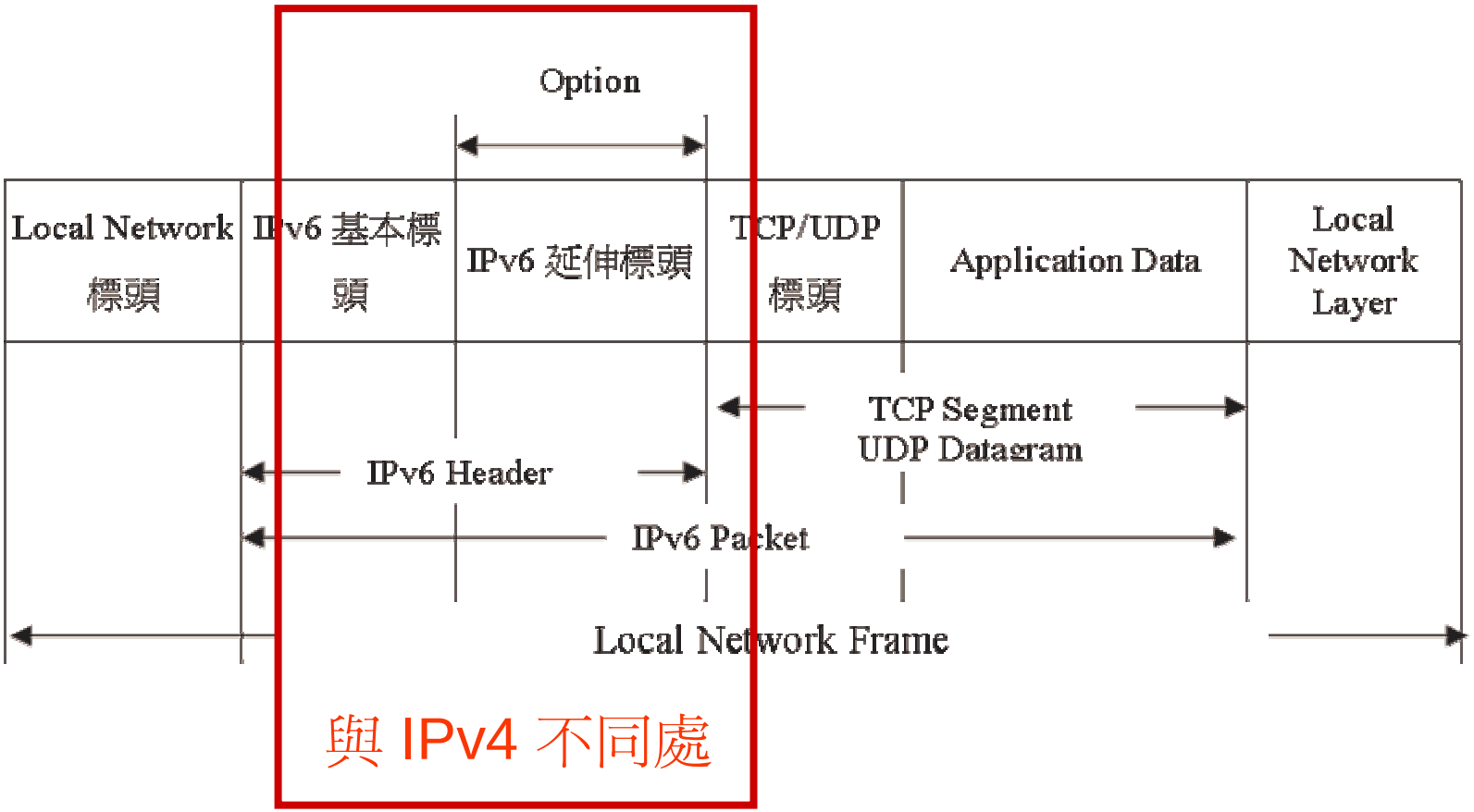
Local Network 標頭



IPv4 傳送架構欄位

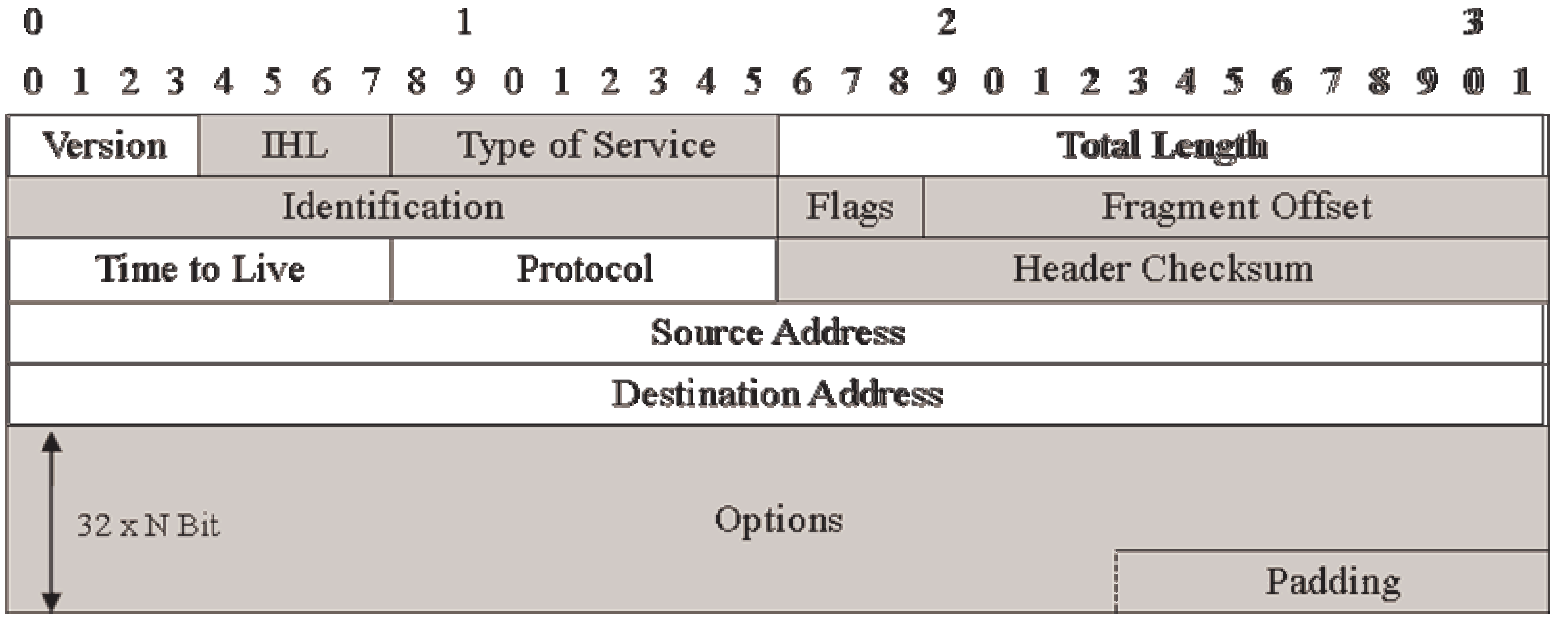
- IPv4 標頭(包含Option)
  - 相當於OSI參考模型中第3層的協定，是TCP/IP最重要的部分。不含Option時，標頭長度為20Byte。包含Option時，即為可變長度。
- TCP/UDP 標頭
  - 相當於OSI參考模型的第4層(Transport層)之Protocol，TCP不含Option時其標頭長度為20Byte，包含Option時，則為可變長度。使用UDP時，標頭長度為8Byte。
- Application Data
  - 上三層中各式的 Service 所產生的資料。
- Local Network Layer Tailer
  - 對應於Local Network 標頭。也就是說，附加於Local Network Protocol之Packet的最尾端，與OSI參考模型的第1層及第2層(物理層及Data Link層)相對應。以Ethernet而言，Tailer為附有4Byte之FCS (Frame Check Sequence)的資料，可確認Frame有無正常運作。

IPv6 Packet 傳送架構



IPv4 Header

- RFC791(Internet Protocol DARPA Internet Program Protocol Specification)



於 IPv6 取消或變更的欄位

IPv4 Header 欄位

- Version (4 bits)
  - 表示Internet Protocol 的版本號碼。IPv4 為0100。
- IHL：Internet Header Length (4 bits)
  - 表示IP 標頭長度的欄位。每32 Bits(4Bytes)為一個單位。
- TOS：Type of Service (8 bits)
  - 指定IP Service品質需求的欄位。但因定義不明確導致相互運用不便，實際上不太被廣泛使用。
- Total Length (16 bits)
  - 以Byte為單位表示封包的總長度，這個長度包含IP 標頭及資料。

IPv4 Header 欄位

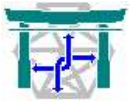
- Identification (16 bits)
  - 記錄被分割(Fragment)的封包，重新組合時的參考資料。
- Flags (3 bits)
  - 這個旗標的最後一個位元是在記錄經分割的封包之後是否還有其他封包存在。
- Fragment Offset (13 bits)
  - 表示被分割的資料，在Datagram中的原始位置。
- Time of Live (8 bits)
  - 記錄封包可以在網路內停留的最長秒數。實際的運用是以通過路由器的台數(Hop Count)來計算的。

IPv4 Header 欄位

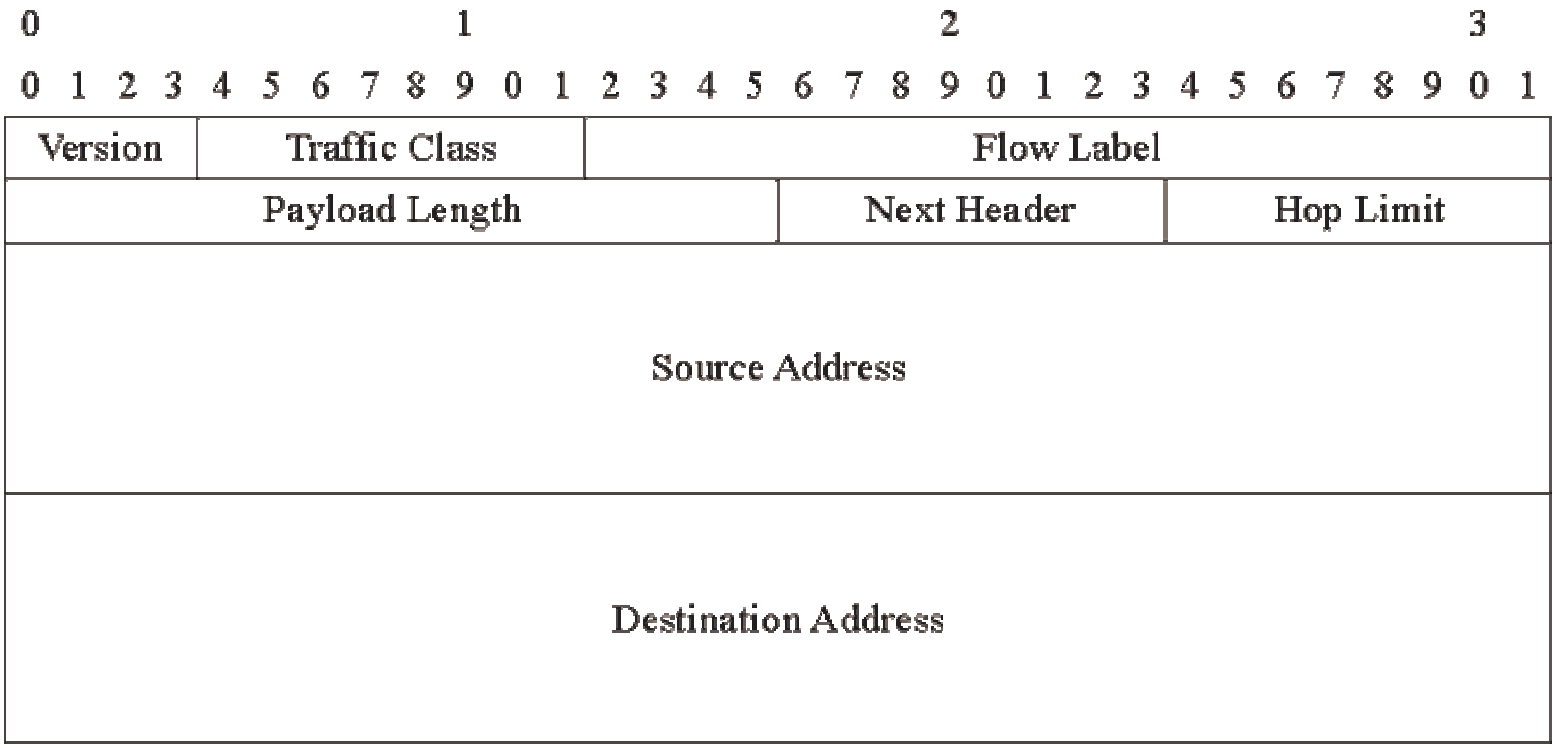
- Protocol (8 bits)
  - 顯示IP的上層(TCP或UDP等)協定的代碼。
- Header Checksum (16 bits)
  - 用來檢查標頭內是否有錯誤用的。
- Source Address (32 bits)
  - 封包來源的IPv4 位址。
- Destination Address (32 bits)
  - 封包目的地的IPv4 位址。

IPv4 Header 欄位

- Option（可變長度）
  - 包含加密等種種附加的服務功能參數都包含在此欄位。
- Padding（可變長度）
  - 此欄位的功用為，使用Option時，當Option 欄位資料的大小不為32 Bits的整數倍時，以0填滿，是其成為32 Bits的整數倍。

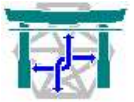


IPv6 Header


0 1 2 3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
Version
Traffic Class
Flow Label
Payload Length
Next Header
Hop Limit
Source Address
Destination Address

Flow Label

- A flow: a sequence of packets which the source desires special handling (by hop-by-hop option or control protocol)
  - identified by combination of source address and non-zero flow label(randomly and uniformly chosen), as a hash key used by router.
  - All packets belonging to the same flow must have the same source address, destination address, priority, and flow label
  - The router may cache the information of next-hop interface, decide how to queue the packet based on its priority, etc…
  - Maximum lifetime of a flow must be specified.



Traffic Class

- The 8-bit Traffic Class field in the IPv6 header is available for use by originating nodes and/or forwarding routers to identify and distinguish between different classes or priorities of IPv6 packets.
  - E.g., used as the codepoint in differentiated services
- General requirements
  - Service interface must provide a means for upper-layer protocol to supply the value of traffic class
  - Value of traffic class can be changed by source, forwarder, receiver
  - An upper-layer protocol should not assume the value of traffic class in a received packet has not been changed.

IPv6 Header 欄位

- Version (4 bits)
  - 表示Internet Protocol 的版本號碼。IPv6 即為0110。
- Traffic Class (8 Bits)
  - 表示封包的類別或優先度。這個欄位與IPv4之”Service Type”提供相同的功能。
- Flow Label (20 Bit)
  - 顯示封包所屬的Flow編號。在不支援Flow Label欄位的機能的主機或路由器上，會使用其預設值0。
- Payload Length (16 Bit)
  - 以無號整數表示在IPv6基本標頭之後剩下的封包長度，以Byte為單位計算。

IPv6 Header 欄位

Next Header (8 bits)

值(10進位)	下一個標頭的種類
0	Hop By Hop Option Header
6	TCP
17	UDP
41	Capsule IPv6 Header
43	Routing Header
44	Fragment Header
46	Resource Reservation Protocol
50	Security Payload Capsule Header (RFC2406)
51	Authentication Header (RFC2402)
58	ICMPv6
59	No Next Header
60	Destination Option Header

IPv6 Header 欄位

- Hop Limit (8 bits)
  - 以無號數表示IPv6 封包被捨棄之前最多可經過的節點數。
- Source Address (128 bits)
  - 封包來源的IPv6 位址。
- Destination Address (128 bits)
  - 封包目的地的IPv6 位址。一般來說，會設定為最終目的地的位址，但若延伸標頭中有Routing Header存在時，則不設定最終目的地，而是設定於Source Routing List所記錄的下一個Route Interface的位址。

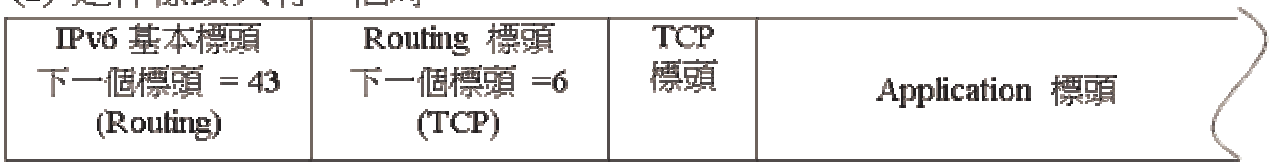
Option的功能

- 並不是所有的IPv6 封包都需使用延伸標頭：
  - 當遇到沒有支援的延伸標頭時，回覆給對方ICMP Parameter Problem Message (Type=1, Code=1)，將封包捨棄。
  - 有延伸標頭，但附有延伸標頭的Option不被支援時，對Option編號要求錯誤處理。

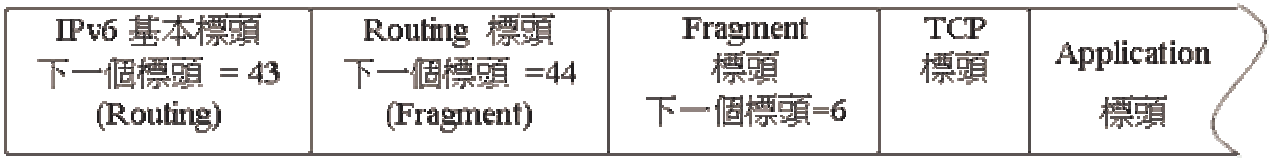
IPv6 封包延伸標頭的例子

(1) 沒有延伸標頭時

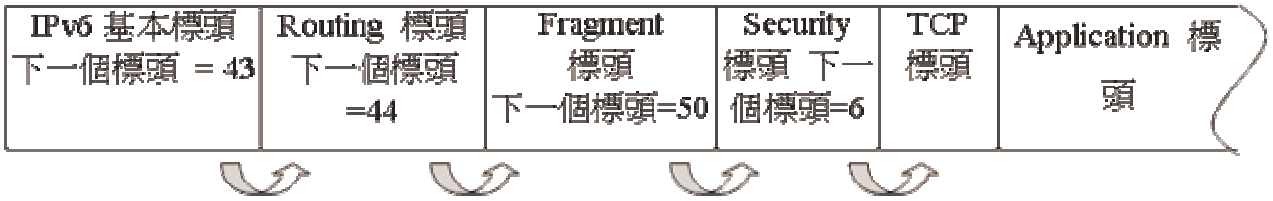
(2) 延伸標頭只有一個時



(3) 延伸標頭只有兩個時



(4) 延伸標頭只有三個時



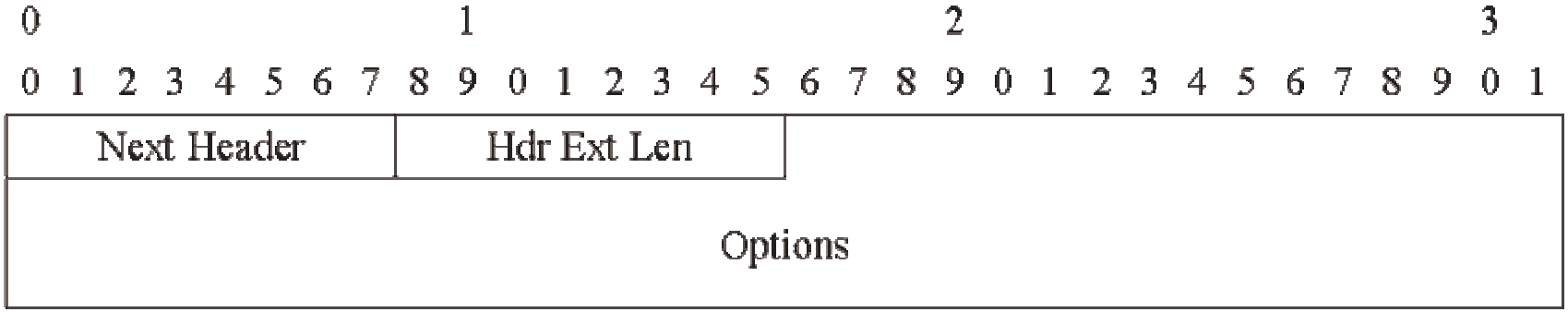
延伸標頭完全實作的標準

- Hop By Hop Option Header(RFC2460)
- Routing Header (Type 0) (RFC2460)
- Fragment Header(RFC2460)
- Destination Option Header(RFC2460)
- Authentication Header (RFC2402)
- Security Payload Capsule (ESP：Encapsulating Security Payload) Header (RFC2406)

延伸標頭出現的順序

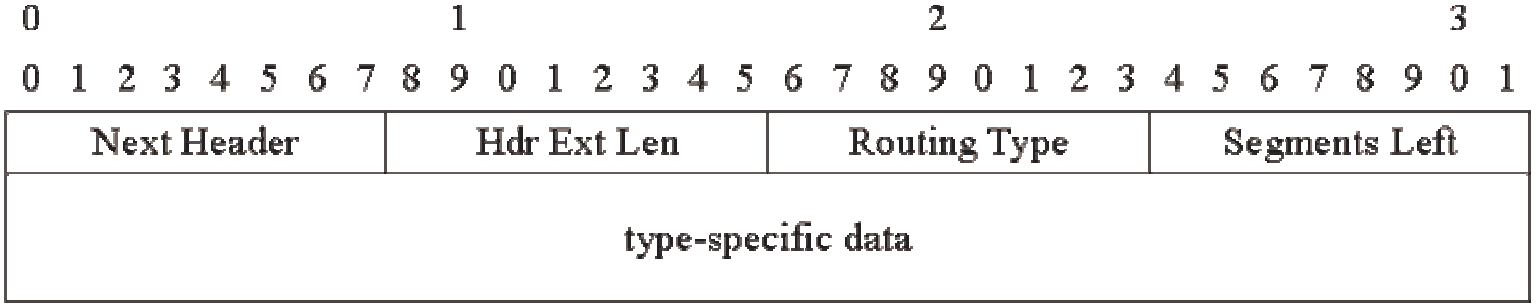
- IPv6 header
- Hop-by-Hop Options header
- Destination Options header
- Routing header
- Fragment header
- Authentication header
- Encapsulating Security Payload header
- Destination Options header

Hop By Hop Option Header



- Next Header (8 bits)
  - 顯示緊接於Hop By Hop Option 標頭之後連續標頭的種類。
- Hdr Ext Len (Header Extension Length) (b Bits)
  - 以無號數來表示Hop By Hop Option 標頭的長度，以8 Bytes為單位。這個長度沒有包含Hop By Hop Option 標頭最初的8 Bytes。
- Options (可變長度)
  - 含一個以上以TLV (Type-Length-Valve)方式編碼的選項。

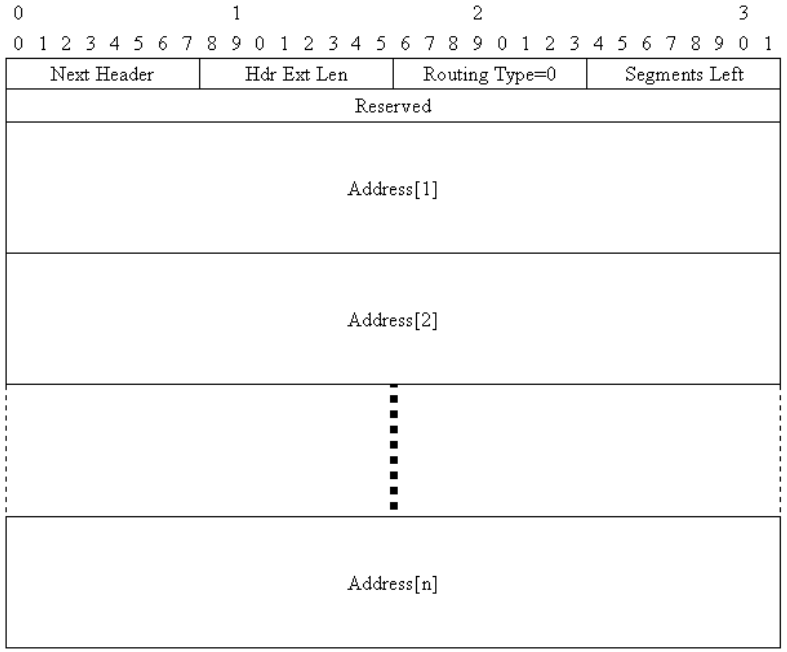
Routing Header



- Next Header (8 bits)
  - 顯示緊接於Routing Header之後連續標頭的種類。
- Hdr Ext Len (Header Extension Length) (8 bits)
  - 以無號數來表示Hop By Hop Option 標頭的長度，以8 bytes為單位。
- Routing Type (8 bits)
  - 表示特定的路由型態。
- Segments Left (8 bits)
  - 以無號數來表示來表示剩餘Segment數，也就是顯示出到達最終目的地必通過之路徑的Segment數。
- Type-Specific Data (可變長度)
  - 此欄位的內容與型式由指定的Routing Type決定。

Source Routing (Routing Type=0)

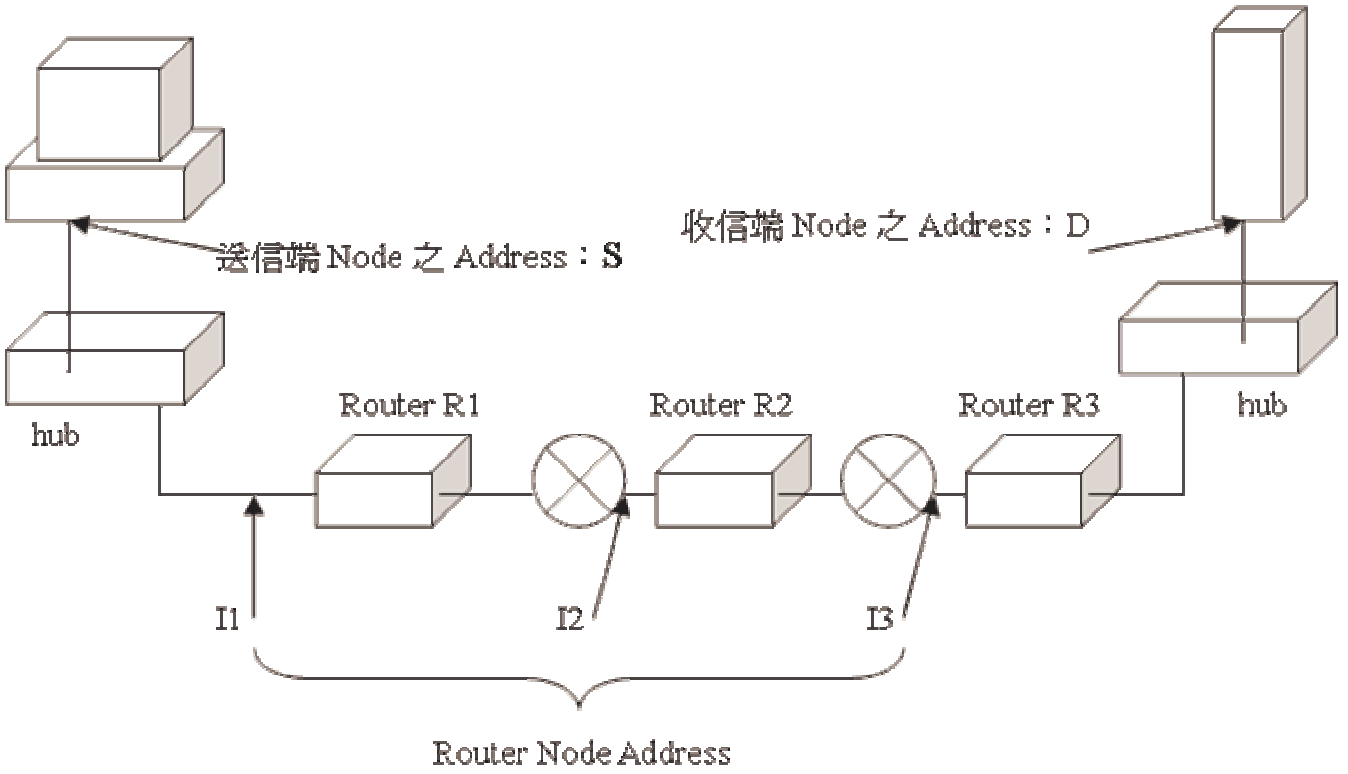
- LSRR (Loose Source and Record Route)
  - 對於路徑上的Segment，會傳達已傳送位址後下一個目的地位址不緊鄰也可以的訊息。
- SSRR (Strict Source and Record Route)
  - 對於路徑上的Segment，會傳達已傳送位址後下一個目的地位址非緊鄰不可的訊息。



Routing Header

- Type 0 routing header
  - For unicast packet only
  - Source places the destination address at the last position of the address list in the routing header
  - Destination address in the IPv6 header is the first (next) desired router on the path
  - The routing header will not be examined until the packet reaches the node identified in the IPv6 header
  - Processing the routing header
    - For strict route, if the first address in the address list is not directly connected, discard the packet
    - Otherwise, swap the destination address of the IPv6 header with address[hdr_ext_len/2-seg_left], decrease the segment left field

Source Routing 的例子



Routing Header Example

- **Source: S, Destination: D Intermediates: I1, I2, I3**
- **As the packet travels from S to I1**

  Source address = S
  Destination=I1
  Hdr Ext Len = 6 (in units of 64 bits)
  Segment left = 3
  Address[1]=I2
  Address[2]=I3
  Address[3]=D

- **As the packet travels from I1 to I2**

  Source address = S
  Destination = I2
  Hdr Ext Len = 6
  Segment left = 2
  Address[1]=I1
  Address[2]=I3
  Address[3]=D

Routing Header Example

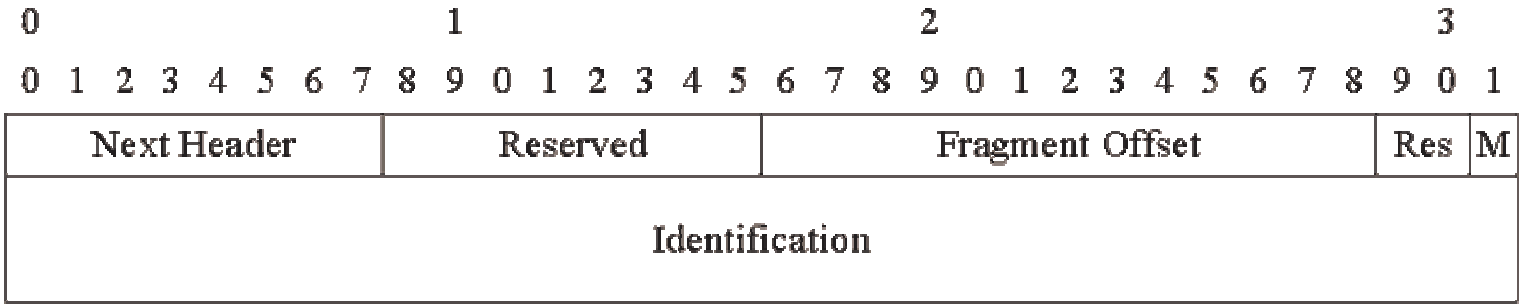
- **As the packet travels from I2 to I3**

  Source address = S
  Destination=I3
  Hdr Ext Len = 6 (in units of 64 bits)
  Segment left = 1
  Address[1]=I1
  Address[2]=I2
  Address[3]=D

- **As the packet travels from I3 to D**

  Source address = S
  Destination = D
  Hdr Ext Len = 6
  Segment left = 0
  Address[1]=I1
  Address[2]=I2
  Address[3]=I3

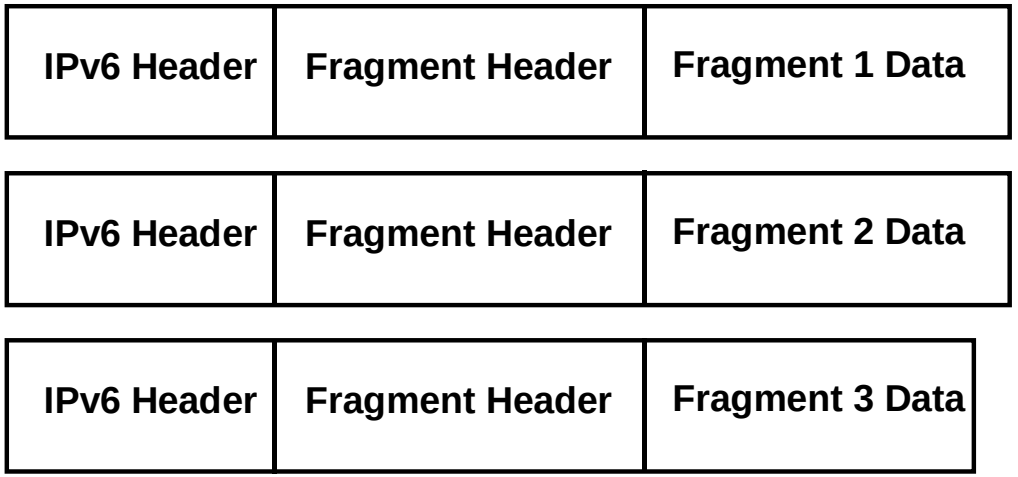
Fragment Header



- Next Header (8 bits)
  - 顯示緊接於Fragment標頭之後連續標頭的種類。
- Reserved (8 bits)
  - 為被保留之欄位。傳送時為0，接收時則忽略掉。
- Fragment Offset (13 bits)
  - 無符號整數。相對於原始標頭之分割部分的起點，跟隨在此標頭之後的資料偏移值，以8bytes為單位。
- M Flag (1 bit)
  - M Flag = 1 尚有Fragment
  - M Flag = 0 最終的Fragment
- Identification (32 bits)
  - 原本附加於各Fragment之識別子，可於Packet再構成時使用。

Fragmentation Example

(a) Original packet



(b) Fragments

Packet Size Issue

- MTU of every link must >= 1280 bytes
  - Use Path MTU Discovery to discover MTU greater than 1280 bytes
  - A node need to accept a fragmented packet that is as large as 1500 octets

Reassembly

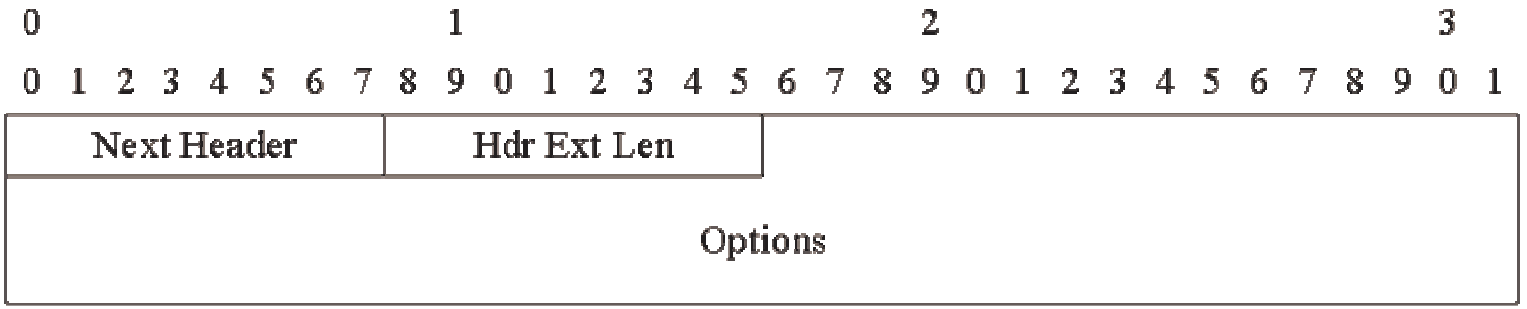
- Unfragmentable part of the reassembled packet consists of all headers up to the fragment header
  - The next header in the unfragmentable part is replaced by the next header field of the first fragment's fragment header
  - Payload length computation

PL.org = PL.first - FL.first - 8 + (8*FO.last) + FL.last

PL: payload
FL: fragment length
FO: fragment offset

Header length up to routing header

Destination Options Header

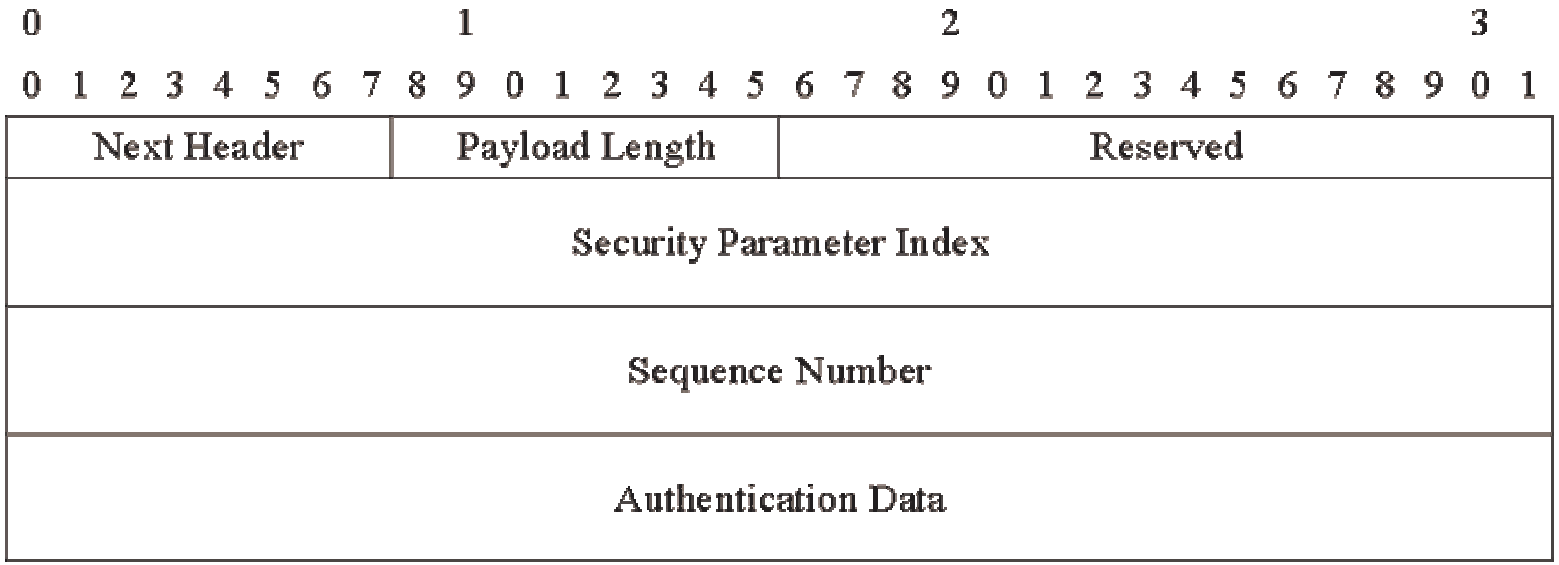


- Next Header (8 bits)
  - 顯示緊接於Destination Options Header之後連續標頭的種類。
- Hdr Ext Len (Header Extension Length) (8 bits)
  - 以無號數來表示Destination Options Header的長度，以8 bytes為單位。這個長度沒有包含Destination Options Header最初的8 bytes。
- Options (可變長度)
  - 此欄位包含一個以上的TLV (Type-Length-Valve)編碼的Option選項。

No Next Header

下一個標頭欄位值為59時，即表示下一個延伸標頭不存在

認證標頭


0 1 2 3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
Next Header
Payload Length
Reserved
Security Parameter Index
Sequence Number
Authentication Data

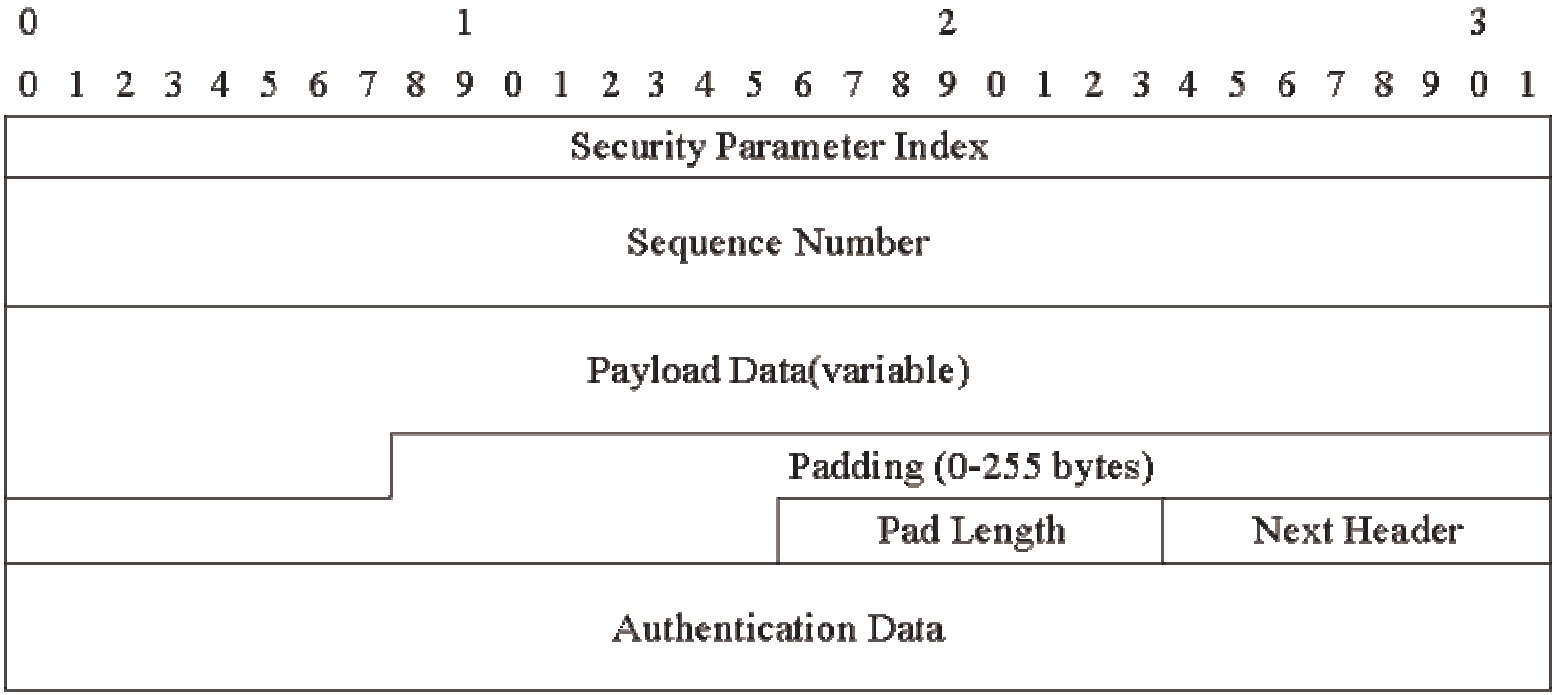
認證標頭

- Next Header (8 bits)
  - 顯示緊接於認證標頭之後連續標頭的種類。
- Payload Length (8 bits)
  - 認證標頭的長度，以4 Bytes(32 bits)單位表示。這個長度沒有包含認證標頭最初的8 bytes。
- Reserved (16 bits)
  - 為被保留之欄位。傳送時為0，接收時則忽略掉。

認證標頭

- Security Parameter Index(SPI) (32 bits)
  - 這個欄位值(SPI值)以32 bits的值將此Datagram之Security Association(SA)指定出來。
- Sequence Number (32 bits)
  - 這個欄位為無號數計數器，是用於防止SA Replay(anti-replay Protection)用的。
- Authentication Data (32 bits 倍數的可變長度)
  - 這個欄位值為以32 bits的整數倍之可變長度，包含為了檢查此封包的安全性之Integrity Check Valve值(ICV)。無法成為32 bits的整數倍時，用Padding來填補。

Security Payload Capsule Header



Security Payload Capsule Header

- SPI：Security Parameters Index (32 bits)
  - 這個欄位以任意之32 bits的值將此Datagram之Security Association SA指示出來之識別子。
- Sequence Number (32 bits)
  - 這個欄位為無號數計數器，是用於防止SA Replay(anti-replay Protection)用的。
- Payload Data (可變長度)
  - 這個欄位值為以32 bits的整數倍之可變長，包含下一個標頭所顯示之Data。此Filed是必要的。
- Padding (可變長 0~255 bytes)
  - 這個欄位是被加密所使用。Payload Data以加密演算法需要之Block Size的倍數被填入。

Security Payload Capsule Header

- Padding Length (8 bits)
  - 計算前一個Padding的長度，以 byte 為單位。數值範圍是0~225，0則表示沒有Padding byte。此欄位是必要的。
- Next Header (8 bits)
  - 顯示緊接於Security Payload Capsule Header之後連續標頭的種類。
- Authentication Data (32 bits 倍數的可變長度)
  - 這個欄位值為以32 bits的整數倍之可變長度，包含為了檢查自此ESP Packet減去認證標頭後計算之Packet的安全性之Integrity Check Valve值(ICV)。無法成為32 bits的整數倍時，用Padding來填補。

參考文獻

[1]Internet Software Consortium: http://www.isc.org/

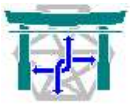
[2] Internet Assigned Numbers Authority: http://www.iana.org/

[3] Internet Engineering Task Force: http://www.ietf.org/

[4] Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy. RFC1519. V. Fuller, T. Li, J. Yu, K. Varadhan. September 1993.

[5] Traditional IP Network Address Translator (Traditional NAT). RFC3022. P. Srisuresh, K. Egevang. January 2001.

[6] Internet Protocol, Version 6 (IPv6) Specification. RFC2460. S. Deering, R. Hinden



參考文獻

[7] Neighbor Discovery for IP Version 6 (IPv6). RFC2461. T. Narten, E. Nordmark, W. Simpson. December 1998.

[8] IPv6 Stateless Address Autoconfiguration. RFC2462. S. Thomson, T. Narten. December 1998.

[9] Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification. RFC2463. A. Conta, S. Deering. December 1998.

[10] IP Authentication Header. RFC2402. S. Kent, R. Atkinson. November 1998.

[11] IP Encapsulating Security Payload (ESP). RFC2406. S. Kent, R. Atkinson. November 1998.