第十章 VPN 網路規劃與管理

10-1 虛擬私有網路簡介

10-1-1 VPN 網路簡介

『虛擬私有網路』(Virtual Private Network, VPN)是網路安全另一個重要的措施，概括而言，它是希望在不安全的**『公眾網路』(Public Network)**上建立一個具安全性較高的『私有網路』。然為何稱之為『虛擬』？是因安全網路是利用軟體或硬體附加在原本不安全網路上，可隨時依其需要建立一個安全通道，使用完畢之後，該安全連線立即消失，並未改變原來的網路架構，故而稱之。

利用 VPN 技術，吾人可將散居世界各地的**『私有網路』(或稱自治系統)**結合成一個安全性較高的網路系統，此網路系統宛如透過防火牆保護的區域網路，此網路稱之為**『VPN 網路』**。在 VPN 網路內各主機可在保護下自由通訊，並讓駭客無法入侵，能符合公司行號全球化的需求。如圖 10-1 所示，某家公司在台北、東京、紐約、曼谷等地方，分別設有據點或工廠，每一地區網路都有防火牆保護，讓外部駭客無法入侵。又透過 VPN 網路結合，讓各地區網路結合成一個安全性高的**『區域網路』**，但它們之間還需透過不安全的**『公眾網路』**連結。如何讓各私有網路之間，透過公眾網路通訊，並能包持它的安全性，則需仰賴**『IP 安全協定』(IP Security Protocol, IPSec)**來達成，亦是本章介紹的重點。

圖 10-1 虛擬區域網路概念

10-1-2 VPN 網路型態

簡而言之，整合各地的區域網路成為一個安全性較高的網路系統，即為『虛擬私有網路』的基本概念。隨著時代的變遷，虛擬私有網路主要有兩種基本型態：WAN-VPN 架構與 Internet-VPN 架構。

(A) WAN-VPN 架構

欲連結各地區域網路成為一個安全性較高的私有網路，最簡單的方法就是向電信公司（如中華電信公司）承租『專線』連接（或稱專屬鏈路），此種網路型態稱之為『廣域網路的虛擬私有網路』（WAN-VPN），如圖 10-2 所示。基本上，電信公司只提供固定連線，沒有路徑選擇功能，並依照傳輸速率與連線距離計費，傳輸速率可介於 64 Kbits 至數百 Gbits 之間。計費方式與傳輸量無關，完全依照傳輸速率與距離計算月租費，如果傳輸距離較近（如圖 10-2，網路之間的地理位置），費率尚可接受，一旦距離過遠（如台北與高雄之間），則月租費已貴得嚇人，更何況跨越國際之間，那幾乎是不可行。再說，WAN-VPN 僅侷限於事先固定的地理位置之間傳輸訊息，無法隨時移動位置。換句話說，

出差人員所到達的地方，除非是架設 VPN 的地區，否則無法與原公司的私有網路通訊。早期為了克服這個問題，大多利用電話撥接來達成，但電話網路傳輸速率慢，而且電話費也很貴，並不完美。由此可見，WAN-VPN 已無法滿足目前國際化的商業環境使用。

但話說回來，WAN-VPN 專屬網路的安全性最高，因為與外界網路完全隔離，閒雜人等不易入侵。因此，WAN-VPN 無需特殊的防護設施，其應用範圍也多侷限於安全防護要求較高的組織單位，如國防部軍事管理的網路系統。

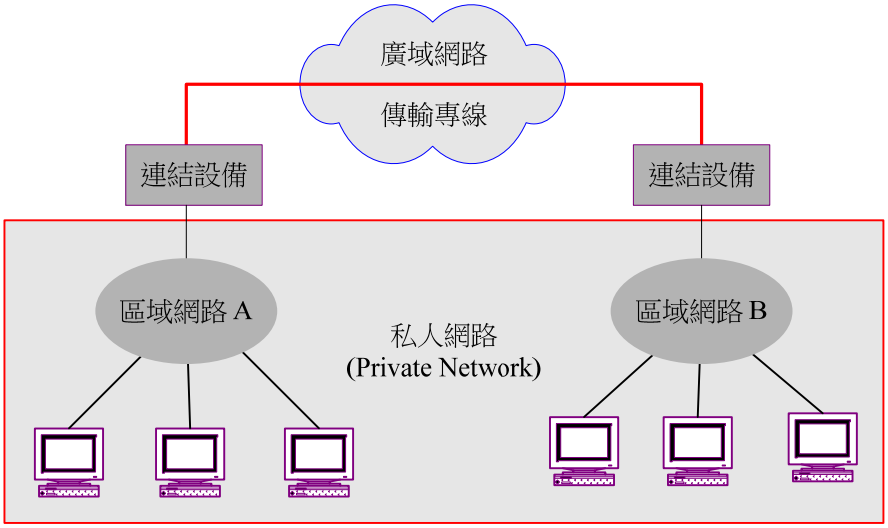


圖 10-2 WAN-VPN 網路型態

(B) Internet-VPN 架構

如前所述，承租沒有路由功能的專線不但價格昂貴，而且也受限於架設位置。”俗擱大碗” 乃是一般人們所欲追求的目標，目前盛行於全球的 Internet 可說是不二人選。利用 Internet 建構 VPN 網路，不但價格便宜，還可藉 Internet 網路的路由功能，將訊息傳送到世界上任何角落，如此一來，所連結的私有網路就不再受地理位置所限。圖 10-2 是利用 Internet 網路所架設的 VPN，可稱之為**『Internet-VPN 架構』**，其中使用**『IP 安全協定』（IP Security, IPSec）**來達成。

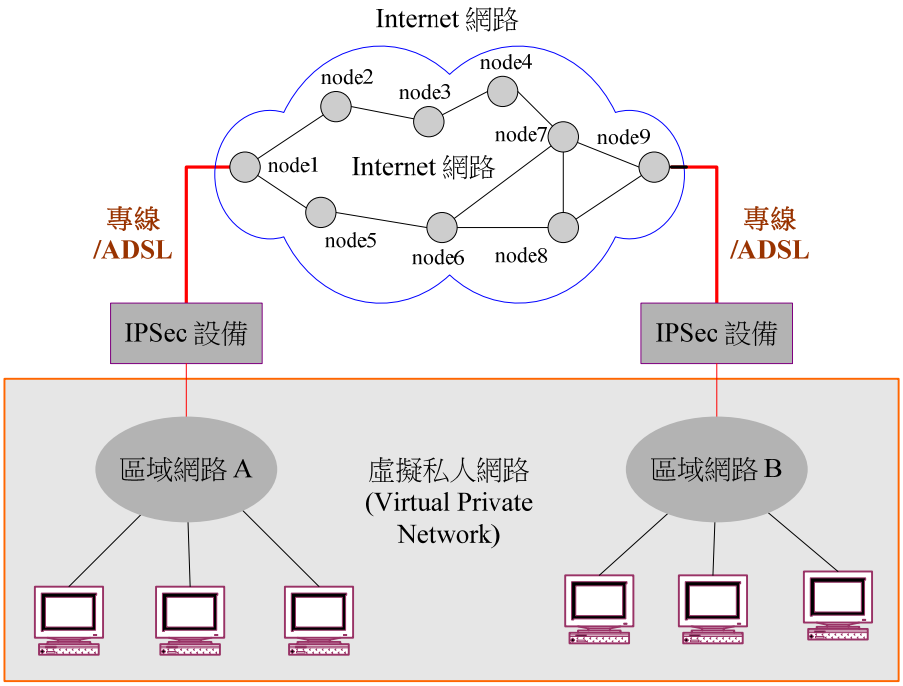


圖 10-3 Internet-VPN 網路型態

VPN 的原理是將圖 10-3 中私有網路對外連接的設備，改換成具有 IPSec 功能的連結設備，就可以達到虛擬私有網路的功能，此連結設備即稱為**『安全閘門』(Secuirty Gateway, SG)**。如圖 10-3 的 VPN 網路成員 (雙方的 SG 閘門)，必須協議雙方可能採用的安全套件 **(包含『鑰匙』)**。其中某一網路有訊息欲傳送到另一網路時，當 SG 閘門收到封包後，則將該封包加入安全措施並重新包裝，譬如訊息加密或認證的處理，之後再將新的封包發送到 Internet 網路上；另一方的 SG 閘門收到封包後，則依照雙方之前所協議的安全套件，將封包回復原來格式，再發送給內部的私有網路。私有網路內工作站發送訊息給另一個工作站時，則不需考慮該工作站是否在本區域網路或其他網路上。攻擊者不了解安全套件內容，或沒有雙方協議的『鑰匙』，也無法盜取或偽造訊息內容。如何協議雙方通訊原則，即是**『IP 安全協定』(IP Security, IPSec)**。

(C) Firewall-VPN 架構

VPN 是藉由 Internet 網路所構成，意指內部網路可能會暴露於 Internet 網路上；攻擊者可能會透過公眾網路來入侵私有網路，因此一般 VPN 網路都必須配合防火牆裝

置，圖 10-4 是一個典型的網路架構。VPN 設備大多安裝在外部路由器上，所以外部路由器除了具備原來封包過濾的功能外，還具備 VPN 的處理能力。譬如，區域網路 A 的使用者想要和區域網路 B 的工作站通訊，它的 IP 封包經由外部路由器(具有 VPN 功能)處理後，再傳送到 Internet 網路上；當區域網路 B 的外部路由器(具有 VPN 功能)收到該封包後，經過適當處理後再轉送給內部網路。另一方面，區域網路如想要和 Internet 上的其他網路通訊，雖不經由 VPN 處理，但也需要依照其安全政策由外部路由器過濾，或經由防禦主機來代理轉送，如此需結合防火牆和 VPN 的功能。也就是說，內部使用者可以選擇是否透過 VPN 處理和外部通訊，VPN 設備也需分辨出所進入的封包是否有經過 VPN 處理，如果有，則表示來自其他所屬機構網路的封包；否則可能是一般外部使用者的訊息。

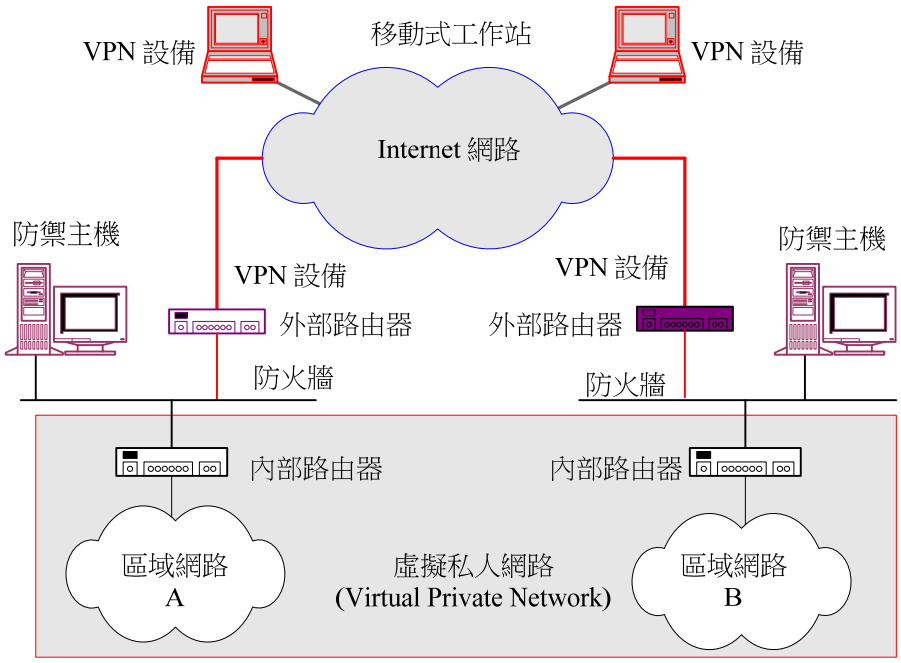


圖 10-4 防火牆型之 VPN 網路

VPN 的另一個重要功能是，許多出差人員或 SOHO 工作人員在外，可能需要連結到公司的私有網路來存取資源，但此類人員所使用的電腦大多屬於客戶端功能，我們只要在其電腦上安裝 VPN 軟體，就可以透過 Internet，並以 VPN 方式連結到公司內部網路。由圖 10-4 可以發現，不管是區域網路 A 或 B，還是移動式工作站的位置，並不

限制其地理位置，只要 Internet 可以到達的地方，都可以建立 VPN 網路，完全合乎企業全球化的需求，至於圖 10-4 的 VPN 設備，目前大多是指具有 IPSec 功能的路由器或主機設備。

10-2 VPN 安全機制

10-2-1 IPSec 與 SSL 協定

在網路安全領域下，有兩個重要的安全協定：**『安全插座協定』(Secure Socket Layer, SSL)** 與**『IP 安全協定』(IP Security Protocol, IPSec)**。如圖 10-5 所示。SSL 協定大多針對應用系統發展的協定，亦是，當 TCP 層建立連線之後，並協議出雙方共用的安全套件，並依照此安全機制下雙方通訊。為了方便發展應用系統，將此安全機制模式建立成標準的函數，讓他如同**『插座』(Socket)** 方便銜接引用 (請參考『資訊與網路安全技術』教材。

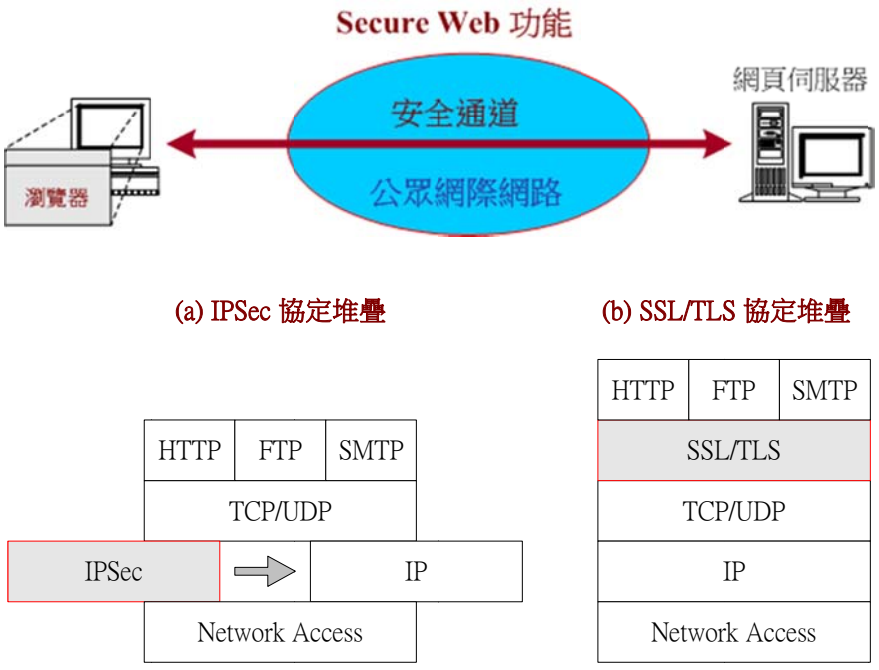


圖 10-5 IPSec 與 SSL 協定堆疊

然而 IPSec 是 ITEF (Internet Engineering Task Force) 特別對 VPN 網路所制定的

安全協定 (規範 RFC 2401)，有分別對 IPv4（IP Version 4）與 IPv6（IP Version 6）制定規範。IPSec 的基本概念不再針對應用系統發展安全機制，而是針對通訊連線在安全機制。Internet 網路上所訊息都是經過 IP 封包封裝後，再以 IP 協定在網路上傳輸。傳送端發送 IP 封包時，並無法預估該封包會經過那些路徑，其間需透過網路上一個接一個路由器轉送始可到達目的地。轉送過程中，每一個路由器收到封包後，由封包上讀取該封包上所註明的目的位址，並尋找可能到達的路徑再傳送出去。如此一來，IP 封包內所承載的訊息很容易被有心人士窺視，或是偽造另一個封包傳送給接收端。由此可見，利用 IP 協定來傳輸資料是非常不可靠的。另一方面，既然所有通訊協定都是利用 IP 協定來傳輸，只要我們能將不可靠的 IP 傳輸，經過安全性機制處理之後，使所承載的任何協定就可達到安全性的保護，換言之，經由 IPSec 協定傳輸的任何應用系統，都可以達到安全性的需求。如圖 10-6 所示，台北網路欲傳送一個封包給紐約網路下某一主機，此封包須經由公眾網路上多個路由器轉送，每一路由器都會拆解封包，並可窺視內容。吾人希望傳送封包須受到安全保護，則需 IPSec 協定是解決 Internet 網路上安全性需求。

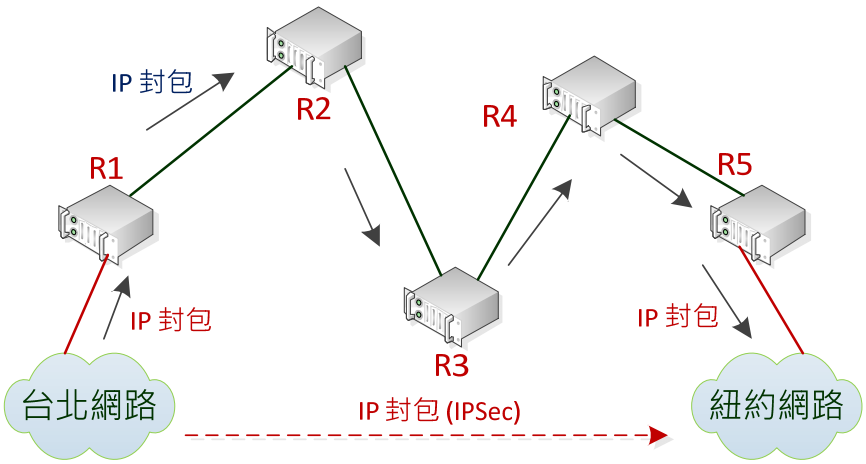


圖 10-6 IPSec 的運作概念

10-2-2 IPSec 相關技術

談到『**安全性**』(**Security**) 總是離不開兩個主題，一則為『**加密**』，其目的是要保

持資料的隱密性，讓他人無法窺視資料的內容；另一則為『認證』，是驗證通訊中的對方身份，是否遭受他人冒名頂替。為了達到上述目的，還是必須仰賴密碼學中加解密演算法，這又牽涉到交換鑰匙的問題。圖 10-5 為 IPSec 的相關技術，我們在這裡先概略性的介紹，讓讀者有一個簡單的概念，接下來再詳細介紹，如此可讓讀者較易進入狀況。

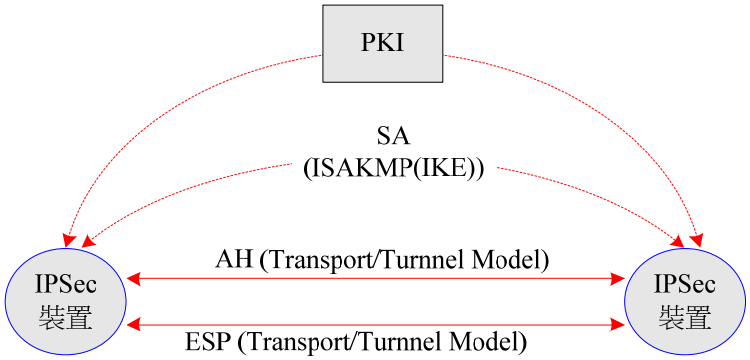


圖 10-7 IPSec 相關技術

由圖 10-7 中，可將 IPSec 相關技術歸納如下：

(1) **IPSec 裝置(IPSec Device)**: 安裝有 IPSec 協定的設備者稱之(或可從事 VPN 功能的設備)，它不僅是一個安全裝置，還可以代表一個使用者個體、使用者群組、或組織單位 (具有身分憑證)。一般 IPSec 裝置設備可分為下列兩種：

- **『安全主機』(Security Host, SH)**: 主機安裝有安全協定者稱之，但必須提供傳輸與通道模式等兩種操作模式 (容後介紹)。
- **『安全閘門』(Security Gateway, SG)**: 路由器安裝有 IPSec 協定者稱之。因為 SG 充當內部網路與外部網路之間的進出閘門，因此，僅提供通道模式。如果使用傳輸模式的話，僅能使用於網路管理協定(如 SNMP 協定)。

(2) **安全關聯 (Security Association, SA)**: 規範通訊實體之間的安全政策，以及某一安全政策之下的相關安全參數，譬如，兩通訊實體之間的安全協定(AH 或 ESP)、封包模式 (傳輸模式或通道模式)、以及加密演算法等等。

(3) **網際網路安全關聯金鑰管理協定(Internet Security Association Key Management**

Protocol, ISAKMP): 通訊實體之間係利用 ISAKMP 協定協調及建立所需的 SA，其協議內容包含安全協定、加密演算法、或認證演算法等等。

(4) **『網際網路金鑰交換』(Internet Key Exchange, IKE)**: 當雙方利用 ISAKMP 協議出所欲採用演算法之後，還必須協議出雙方的會議金鑰，此鑰匙可能使用於加密或認證系統。IPSec 為了使 ISAKMP 能符合各種環境需求，並不固定某一特定的金鑰交換協定，而由另一個 Internet 網路上較普遍的 IKE 協定來完成。

(5) **公鑰基礎架構 (Public Key Infrastructure, PKI)**: PKI 發給每一個 IPSec 身份驗證的數位憑證，作為進入 VPN 網路的身份證明，其中包含個體的公鑰 (Public Key) 與私鑰 (Private Key)。通訊實體之間就是利用 PKI 所發給的鑰匙互相確認身分，並交換鑰匙材料以建立會議金鑰。相關技術請參考第九章介紹。

(6) **認證標頭 (Authentication Header, AH)**: 認證標頭是 IPSec 的兩種安全協定之一。IPSec AH 主要認證封包標頭是否有遭受竄改或偽裝，其中有『傳輸模式』與『通道模式』兩種封包模式。

(6) **封裝安全承載 (Encapsulation Security Payload, ESP)**: ESP 是 IPSec 的另一個安全協定。IPSec ESP 將原 IP 封包經過加密後，重新封裝成另一個 IP 封包，以達到資料隱密性的功能，同樣也有『傳輸模式』與『通道模式』兩種封包模式。

(7) **操作模式 (Operating Mode)**: IPSec 協定有『傳輸模式』(Transport Mode) 與『通道模式』(Tunnel Mode) 兩種操作模式，無論 AH 或 ESP 協定都可以使用這兩種操作模式來傳輸訊息；因此，IPsec 協定有四種訊息封包格式，如圖 10-6 所示。

(8) **演算法 (Algorithm)**: 無論認證 (Authentication) 或加密 (Encryption) 都需要相對應的演算法。基本上，IPSec 並不規定標準演算法，而是雙方利用 ISAKMP 協定協議而成。

	IPSec AH	IPSec ESP
Transport Mode	AH with Transport Mode	ESP with Transport Mode
Tunnel Mode	AH with Tunnel Mode	ESP with Tunnel Mode

圖 10-8 操作模式

有了上述相關技術之後，接著來探討它們之間的關聯性，如此可讓讀者稍微瞭解 IPSec 的運作概念，至於詳細的運作程序將會在相關協定中說明，簡述如下：

(1) IPSec 協定包含 IPSec AH 與 IPsec ESP 兩種安全協定，這兩種安全協定都有傳輸模式和通道模式等兩種封包格式；

(2) 至於通訊雙方是要採用何種安全協定及封包格式？視安全關聯（SA）的規範而定

(3) 如何制定 SA 的安全規範？係由通訊雙方利用 ISAKMP 協定所協議完成的；

(4) 在 ISAKMP 協議當中若需交換鑰匙作為身份確定或制定會議金鑰，可利用 IKE 協定來完成；

(5) 在雙方認證身分或交換鑰匙時，必須有代表身份的公鑰，然而此公鑰可由 PKI 系統中的憑證授權（CA）中心發給。

10-2-3 IPSec 運作程序

在 VPN 網路下，兩個端點的『安全閘門』之間協議出 IPSec 安全機制，再利用此機制下 IPSec 封包互相傳送訊息。但 IPSec 封包進入公眾網路之後，便如同一般 IP 封包備層層轉送到目的地，封包傳送途中還是會被盜取窺視，但封包內容受到保護(加密)，他人無法知曉其內容。其運作程序如圖 10-9 所示，說明如下：

(1) 區域網路內 A 工作站，欲傳送訊息給區域網路 B 的工作站 B，它將 IP 封包發送到網路上；

(2) 安全閘門 SG_A 收到封包後，由封包標頭得知是 VPN 的目地位址。則由 SAD (SA Database) 搜尋是否有相關安全關聯 (SA) 可用，如果有則立即發送 IPSec 封包；

(3) 如果沒有 SA 則啟動 ISAKMP 協定，協議雙方安全套件，包含： IPSec AH 或 IPSec ESP、身分認證、訊息確認與密碼系統等等；並啟動 IKE 作雙方協議動作。

(4) IKE 不僅實現雙方協議事項(兩階段協議)，並計算密碼系統內所需的鑰匙，如加密鑰匙與訊息確認鑰匙。

(5) 將雙方所協議成功的密碼套件儲存於 SAD 內，以備下次使用。

(6) SG_A 與 SG_B 之間協議或取得 SA 之後，便依照 SA 內規範傳輸。首先 SG_A 將收到的 IP 封包包裝成 IPSec 封包(IPSec AH 或 IPSec ESP)，再將其發送到 Internet 網路上。

(7) IPSec 封包就如同一般 IP 封包一樣，在多個路由器轉送之下到達 SG_B。IPsec 封包內訊息受到安全包護，他人無法窺視或竄改其內容。

(8) SG_B 收到 IPSec 封包後，再依照雙方協議的 SA 套件，將 IPSec 封包恢復原來 IP 格式，再發送到區域網路 B 內。

(9) 工作站 B 收到該封包，與原來工作站 A 發送的完全相同。

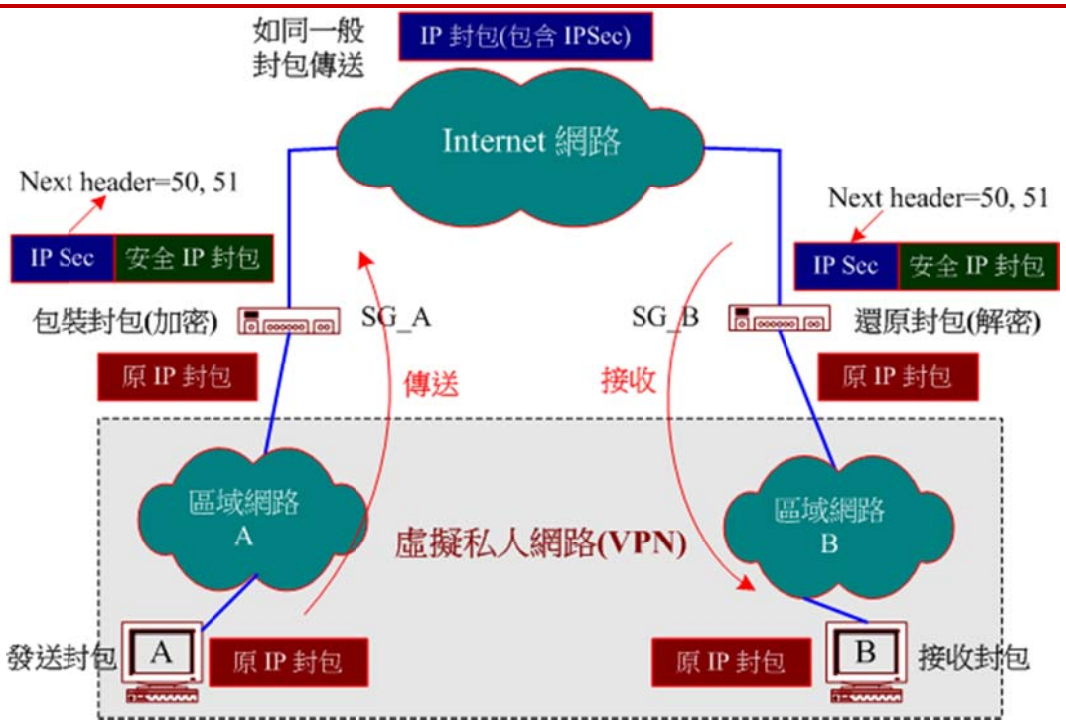


圖 10-9 協議雙方安全機制

乍看之下，IPSec 好像很複雜的樣子，這是因為它必須結合許多安全措施(如 PKI、ISAKMP、IKE) 才能達成。在此假設每一參與 VPN 運作者都已取得數位憑證 (有關憑證認證與身份識別的議題，請參考『資訊與網路安全技術』)。首先介紹 AH 與 ESP 安全協定的運作程序，接著再介紹 SA 的安全參數；至於如何利用 ISAKMP 協定建立 SA，與協議產生雙方的會議金鑰的 IKE 協定，也請參考『資訊與網路安全技術』。

10-3 IPSec AH 安全協定

10-3-1 IPSec AH 協定簡介

『認證標頭』(Authentication Header, AH) 是 IPSec 最基本的安全協定，它是針對 IP 封包標頭做安全認證，提供有『非連接導向的完整性』、『資料來源認證』、以及『反重播』等保護服務。另外 AH 並未對所承載的資料作任何保護，基本上，IP 封包在 Internet 網路上，乃經由路由器 (或網路閘門) 層層轉送才會到達目的地，每經過一個路由器轉送時，路由器便拆解該封包的標頭，根據標頭上所標示的目的位址，繼續往下一

個路徑傳送；當然每一路由器都會重新包裝該封包，並製作新的封包標頭，所以有心人士非常容易去竄改封包標頭，或從事重播攻擊的行為。AH 的功能是對 IP 封包提供認證，確保遭受竄改的封包可以被偵測出來。

AH 使用密碼學的『**訊息認證碼**』(**Message Authentication Code, MAC**) 來認證 IP 封包標頭。簡單的說，傳送端將 IP 封包標頭經過雜湊演算法得到一個訊息摘要(Message Digest, MD)，再將此訊息摘要經過秘密金鑰加密，得到一個 MAC 碼，最後將此 MAC 碼 (製作 AH 標頭) 與 IP 封包一併傳送給接收端；接收端收到此封包後，以同樣的演算法與秘密金鑰產生另一個 MAC 碼。如果兩者 MAC 碼相同的話，表示封包未遭受竄改或偽造；否則需拋棄此封包。目前最常用的 MAC 演算法是 **HMAC (Hash Message Authentication Code)**，主要原因是，它可以配合不同的雜湊演算法，譬如，MD5、SHA-1、RIPEMD-160 或 Tiger 等雜湊函數。另外加密時所需的秘密金鑰是在安全關聯 (SA) 裡所制定，在此暫時不去討論它如何產生。

(A) AH 標頭格式

認證標頭 (AH) 是 IPSec AH 協定所產生的一個新認證標頭，主要功能是認證原封包標頭是否遭受竄改，並將它置放於原封包標頭的後面。AH 標頭包含五個固定長度的欄位和一個不定長度的認證資料欄位，如圖 10-10 所示，各欄位功能如下：

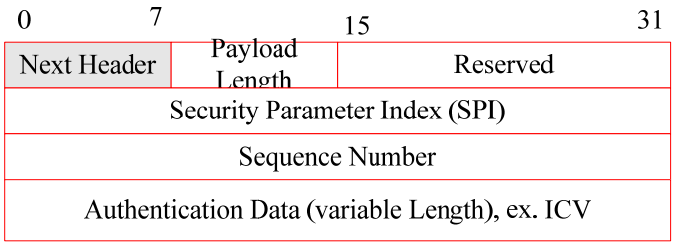


圖 10-10 認證標頭格式

- **下一個標頭 (Next Header, NH)**：8 位元欄位。標示 AH 標頭後面緊接著的封包格式。譬如，TCP 封包為 6；IP 包裝的 IP 封包是 4。如果後面緊接著 ESP

標頭，則為 50，而 AH 的編號是 51。

- **承載長度（Payload Length）**：8 位元欄位。表示整個 AH 標頭欄位的長度，計算方式是以 32 位元字組為單位，再除以 2，因為 IPv6 封包標頭是以 64 位元字組為單位。

- **保留（Reserved）**：16 位元保留欄位。目前皆設定為 0，還未指定使用方式。

- **安全參數索引（Security Parameter Index, SPI）**：32 位元欄位。SPI 是通訊雙方事先協議完成的安全關聯(SA)索引值，它必須配合目的位址和安全協定(AH 或 ESP) 來查詢相關安全參數。

- **序號（Sequence Number）**：32 位元的無號整數值，是一個計數器的數值，主要防止重播攻擊（Replay Attack），容後再介紹。

- **認證資料（Authentication Data）**：不定長度的欄位。此欄位所存放的便是原來 IP 封包標頭的『完整性檢查值』(Integrity Check Value, ICV)，即 IP 封包標頭經過 HMAC 運算後的值。如果是 IPv4 封包，則 ICV 的長度必須是 32 的整數倍，而 IPv6 必須是 64 的整數倍。AH 協定規定 IPSec 裝置至少必須提供 HMAC-MD5 與 HMAC-SHA-1 等兩個以上的 HMAC 演算法。

(B) AH 認證欄位

AH 認證欄位是傳送端選擇原來 IP 封包標頭上某些欄位的值，並將這些值經過 MAC 演算法計算，產生一個『完整性檢查值』(Integrity Check Value, ICV)，再將此 ICV 值存放於 AH 標頭的認證資料欄位（如圖 10-10 所示）上；接收端收到 IPSec AH 封包後，選擇同樣欄位計算出 ICV，如果該 ICV 與 AH 標頭上 ICV 相同的話，則表示該封包是正確的，但必須滿足下列三個需求：

1. 必須協調出針對 ICV 加密的秘密金鑰；

2. 必須協調出採用何種認證演算法（如 HMAC-SHA-1）；

3. 必須協調出選擇哪些欄位來計算 ICV 的值。

基本上，這三個需求都必須在通訊之前，透過 SA 連線協議而成。但就第三個需求而言，除了雙方可協議出採用哪些欄位外，我們同時必須了解選擇欄位的原因。有些欄位的內容會隨時改變，並不適合做 AH 認證使用，否則會發生許多無謂的困擾。如以 TTL 與 Header Checksum 欄位為例，IP 封包每經過一個路徑（或路由器），則 TTL 的值便會被減一，之後路由器會再重新計算標頭檢查值（Header Checksum），因此這兩個欄位隨時會遭受修改其內容。如果傳送端將隨時變更欄位的值加入計算的話，接收端在做認證檢查時，將很困難去辨別是正常變更或是遭受破壞。因此，我們必須先了解 IP 封包標頭上有哪些欄位容易變更、以及哪些欄位較不容易變更。在協調雙方通訊參數時（SA 連線），便可參照這些訊息來決定哪些欄位可加入認證範圍。當然，加入愈多的欄位則認證的安全性愈高，但這必須視通訊雙方的需要而定。

吾人將 IPv4 封包標頭較容易變更的欄有位：服務類別(Type of Service, TOS)、旗號（Flags）、存活時間（Time To Live, TTL）、分段偏移位址（Fragment Offset）、標頭檢查碼（Header Checksum）、選項（Options）。

不容易被變更欄位並可以選擇 ICV 計算欄位：

- **版本（Version）：** IP 封包版本，應該為 4。
- **Internet 標頭長度（Internet Header Length, IHL）：** 封包標頭的長度（包含 Options 與 Padding 欄位）
- **總長度（Total Length）：** 封包的總長度，其中包含封包標頭與承載資料（Data 欄位）。
- **識別（Identification）：** 如果所承載的資料（如 TCP 封包）有經過分段後，再

分別由不同的 IP 封包承載，則此欄位登錄該資料的分段號碼。

◈ **通訊協定（Protocol）：**表示封包所承載資料是屬於何種通訊協定，譬如 TCP、UDP、或 ICMP 等等。

◈ **來源位址（Source Address）：**封包的來源位址。

◈ **目的位址（Destination Address）：**封包的目的位址。

基本上，選擇那些欄位計算 ICV，是由雙方協議的 SA 來決定，我們將 IPv4 的封包標頭顯示於圖 12-12，其中有底色的欄位表示有可能被選取的機會。如果沒有被選取的欄位，在計算 ICV 時都會被設定為零。

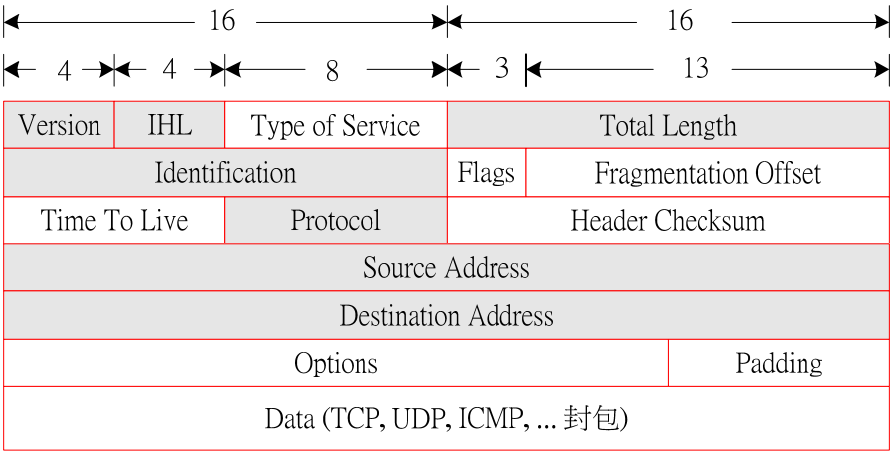


圖 10-10　IPv4 標頭可參與計算 ICV 欄位（有陰影部份）

我們用兩個簡單的範例，說明選擇那些欄位計算 ICV，可能會對 AH 認證能力產生影響。一者將總長度欄位加入計算，倘若傳輸當中封包所承載資料遭受替換，雖然攻擊者也可以修改總長度欄位的內容，來蒙騙接收者；但接收端還是可以由 ICV 計算出認證封包長度是否被變更，如此一來，表示 IPSec AH 不但可以保護封包標頭，也可認證整個 IP 封包，此即稱為『非連接導向的完整性』功能。另一者，倘若將來源位址加入 ICV 計算，則可以避免中間人攻擊，其原因是中間人將封包接收後，再發送新的封包給接收端，則新的封包的來源位址勢必遭受變更，接收端便可依此驗證出該封包的正確性，此

即稱為『資料來源認證』功能。

10-3-2 IPSec AH 操作模式

IPSec AH 協定有：傳輸模式與通道模式等兩種操作，說明如下：

（A）AH 傳輸模式

AH 傳輸模式是將認證標頭放置於 IP 封包標頭與傳輸層協定（TCP、UDP 或其他協定）的標頭之間，有 IPv4 與 IPv6 兩種不同封包格式。圖 10-11 (a) 為原 IPv4 封包，經過 AH 傳輸模式包裝後的格式如圖 10-11 (b)所示，IP 標頭的長度可以由 20 Bytes 到 60 Bytes 之間，之所以不定長度是因為有可能在標頭後面加入選項（Options）資料，如果沒有選項資料，則 IP 標頭長度為 20 Bytes。

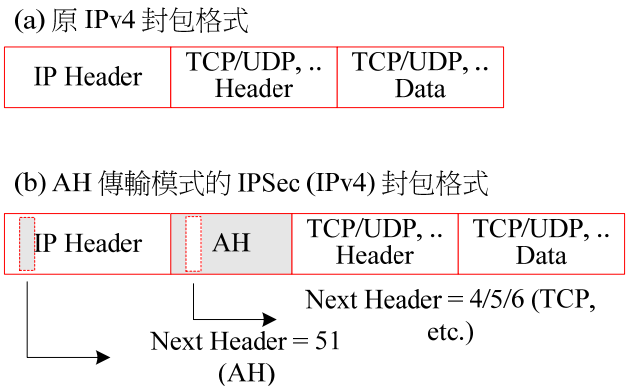


圖 10-8 IPv4 的 AH 傳輸模式包裝

基本上，原來 IPv4 封包標頭是不用修改的，只要將 AH 標頭加入即可，但因原 IP 封包所承載的協定已變成 AH 協定，而非原來的協定，因此還是需要將 IP 標頭內的『協定』（Protocol）欄位內容設定為 51（AH 協定），並將原來的值置於 AH 標頭的『Next Header』欄位上。譬如，原來 IP 封包所承載的是 TCP 協定（Protocol=5），經過 AH 傳輸模式包裝後，IP 標頭上的 Protocol 欄位便成為 51（AH 協定），而 AH 標頭上的 Next Header 欄位則需設定為 5（TCP 協定，如圖 10-8 (b) 所示）。

（B）AH 通道模式

所謂『**通道模式**』(**Tunnel Mode**)，即是隱藏原來的 IP 標頭，而另外製作一個新的封包標頭，並且利用 AH 標頭保護原來的 IP 標頭。圖 10-12 為 IPv4 與 IPv6 AH 通道模式的封包包裝，新的封包標頭稱之為『**外部標頭**』(**Outer Header**)；而原來封包標頭稱之為『**內部標頭**』(**Inner Header**)。基本上，外部標頭的封裝格式與原 IP 標頭一樣，但它的目的位址與來源位址，可能和內部標頭不同，不相同的原因是該 AH 通道所扮演的角色有所不同(連接主機或安全閘門，容後介紹)。外部標頭的內容也可以被所經過的路由器修改，譬如 TTL 或標頭檢查碼等欄位。

(a) AH 通道模式的 IPSec (IPv4) 封包格式

New IP Header	AH	Original IP Header	TCP/UDP, .. Header	TCP/UDP, .. Data

(b) AH 通道模式的 IPSec (IPv6) 封包格式

New IP Header	New Extesion Header	AH	Original IP Header	Orig Extesion Header	TCP/UDP, . . Header	TCP/UDP, . . Data

* 假設 Extension Header 存在

圖 10-12 IPv4 與 IPv6 AH 通道模式的封裝格式

10-4 IPSec ESP 安全協定

10-4-1 IPSec ESP 協定簡介

IPSec 另一個安全協定為『**封裝安全承載**』(**Encapsulation Security Payload, ESP**)，同樣包含 IPv4 與 IPv6 兩種規範。ESP 協定是將原來封包所承載的資料經過加密處理之後，再重新封裝一個新的封包(ESP 封包)才傳送給接收端；接收端拆解 ESP 封包後，先將資料解密，再組合回原封包格式，故稱之為『封裝安全承載』，其特性歸納如下：

(1) ESP 提供資料的隱密性、資料來源認證、非連接方式完整性、反重播攻擊能力、以及有限度的流量機密性等功能。

(2) 具有傳輸模式（Transport Mode）和通道模式（Tunnel Mode）等兩種封包格式。

(3) 利用封包序號作為防禦重播攻擊（如同 IPSec AH）。

(4) 對所承載的資料可進行加密，以達到隱密性功能。一般都採用對稱加密法，針對加/解密所需的秘密金鑰，以及加密演算法皆是由 SA 參數而定。基本規範有 DES 的 CBC 模式與 NULL 編碼演算法。

(5) 可利用 ICV 驗證整個封包資料，以達到資料來源驗證。認證的範圍（亦是所選用的欄位）可由雙方協議的 SA 參數而定，認證演算法有 HMAC-MD5、HMAC-SHA-1 以及 NULL。

(6) 必須採用通道模式才具『有限度的流量機密性』的功能。

(7) 可配合 AH 協定使用，以達到較完整的封包標頭驗證，與資料隱密性的功能。

相較於 AH，ESP 好像增加了資料隱密性和有限的流量機密性功能，但對於資料來源的認證，就沒有 AH 協定那麼完整。簡單的說，AH 協定主要是提供封包標頭的認證，任何有修改或偽裝封包將被偵測出來，但對於資料是否認證功能，完全取決於所選用的認證欄位而定；當然 ESP 協定除了提供簡單的封包標頭認證之外，並將所承載的資料加密，以達到資料隱密性的功能。

(A) ESP 封包格式

ESP 封包格式與 AH 有很大的不同點，AH 是將認證標頭插入原封包標頭的後面或前面，基本上還是保留原來的封包格式；然而 ESP 為了達到資料的隱密性，會將原封包所承載的資料重新包裝成另一個『ESP 封包格式』，並依照操作模式（傳輸或通道模式）處理原封包標頭。譬如，傳輸模式就是將 ESP 封包放置於原封包標頭的後面；而通道模式是新建立一個的封包標頭，放置於最前面（容後詳細介紹）。

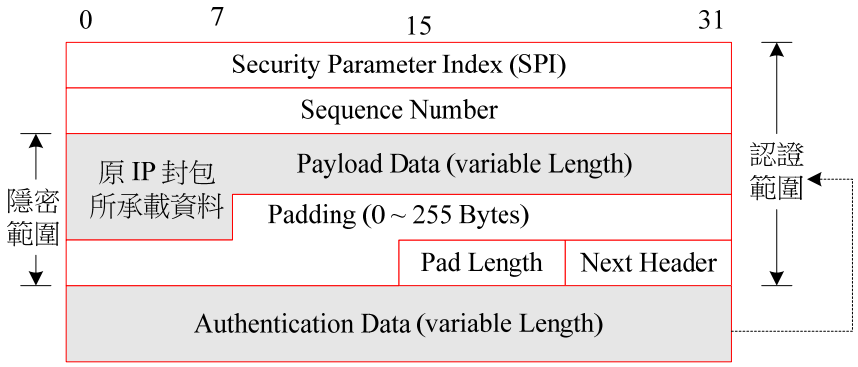


圖 10-13 IPSec ESP 封包格式

圖 10-13 為 ESP 封包格式（未包含原封包標頭），它是將原封包所承載資料經過加密(或未加密)後，再重新包裝的格式。一般將 SPI 與 Sequence Number 稱之為『ESP 標頭』(ESP Header)，而 Payload Length 與 Next Header 兩個欄位稱之為『ESP 標尾』(ESP Trailer)，各欄位功能如下：

- **安全參數索引（Security Parameter Index, SPI）**：32 位元長度。SPI 的功能和 AH 協定中的 SPI 相同，都是雙方事先協議完成安全關聯（SA）的索引值，但它必須配合目的位址和安全協定（AH 或 ESP）來查詢相關安全參數。

- **序號（Sequence Number）**：32 位元長度。如 AH 協定中的序號一樣，都是作為反重播攻擊使用。

- **承載資料(Payload Data)**：為不定長度的欄位。當有啟動隱密性資料功能時(SA 參數決定)，ESP 協定先將原來 IP 封包所承載的資料（TCP、UDP 或其他協定)，經過某一加密演算法編碼後，再存入此欄位上傳送；如果沒有啟動隱密性功能，則填入原來 IP 所承載的資料。另一方面，如所選用的加密演算法需要『初始向量』(Initial Vector, IV) 的話，則必須將 IV 值填入此欄位。基本上，無論是只有密文或密文附帶 IV 值，資料都必須是一個可以被 8 整除的長度。

- **填補(Padding)**：此欄位是選項的不定長度。主要作用於對齊資料長度是否是 32 位元長的整數倍，但它的長度可以由 0 到 255 位元組。

- **填補長度(Pad Length)**: 8 位元長度。此欄位是表示所加入的填補資料的長度，有效值是 0 ~ 255 之間。

- **下一個標頭（ Next Header)**: 8 位元長度。此欄位是用來辨識封包裡所承載資料的協定，譬如，Next Header = 6，表承載資料為 TCP 協定的資料封包。

- **認證資料（ Authentication Data)**: 不定長度欄位。此欄位所存放的是『完整性檢查值』(ICV)，認證範圍可以由 SPI 到 Next Header 欄位，這可由雙方協議的 SA 參數而定。

由上述的介紹，可以分辨出 AH 和 ESP 兩協定之間最大的不同點，AH 協定較著重於封包標頭認證，因此對於封包來源認證功能較強；然而 ESP 協定則偏重於承載資料的隱密性及認證，對於資料的保護較為嚴密。使用者可依照環境需求選擇 AH 或 ESP 協定，甚至也可以整合 AH 與 ESP 協定使用。

(B) ESP 加密及認證演算法

基本上，ESP 都使用對稱加密演算法。這是因為公開金鑰演算法必須耗費較長的加密/解密時間，這對於一般通訊而言效率太低。另一方面，IPSec 只建議 VPN 系統至少需具備有 DES 與 NULL 兩種編碼演算法，其中 NULL 表示所承載的資料是沒有經過編碼的，亦即選用 NULL 編碼演算法，則表示沒有加密的功能。除了上述兩種編碼系統外，通訊雙方也可以經由 SA 連線來協議雙方的編碼系統（如 AES 演算法)。

同樣的，IPSec 並沒有強制規定一定要用何種認證演算法，但規定至少要有：HMAC-MD5、HMAC-SHA-1 與 NULL 等演算法，其中 NULL 表示沒有認證功能的意思。無論所採用的加密系統、驗認演算法或演算法中所需的秘密金鑰，都必須雙方經由 ISAKMP 協定協議出來，如果在協議之中需要交換雙方鑰匙，就會使用到 IKE 協定。

10-4-2 IPSec ESP 操作模式

IPSec ESP 協定有：傳輸模式與通道模式等兩種操作，說明如下：

（A）IPv4 ESP 傳輸模式

如同 AH 協定一樣，ESP 協定也分為『傳輸模式』與『通道模式』兩種操作模式，其最大的不同點在於 ESP 標頭（SPI 與 Sequence Number 欄位）所存放的位置，以及是否重新建立新的封包。以下分別就 IPv4 與 IPv6 來介紹這兩種運作模式。

圖 10-14 為 IPv4 ESP 傳輸模式的封包格式，它是將 ESP 標頭（SPI 與 Sequence Number 欄位）插在原 IP 封包標頭之後，並對原封包所承載的資料編碼加密，再存放於 ESP 承載欄位上；緊接著是 ESP 標尾（Padding、Pad Length 與 Next Header 等欄位），最後才是 ESP 認證資料的欄位（ICV 資料）。其中經加密編碼欄位是否包含 ESP Trailer、或所提供的認證範圍（ESP 標頭到 ESP 標尾之間），皆由雙方協議之 SA 而定。

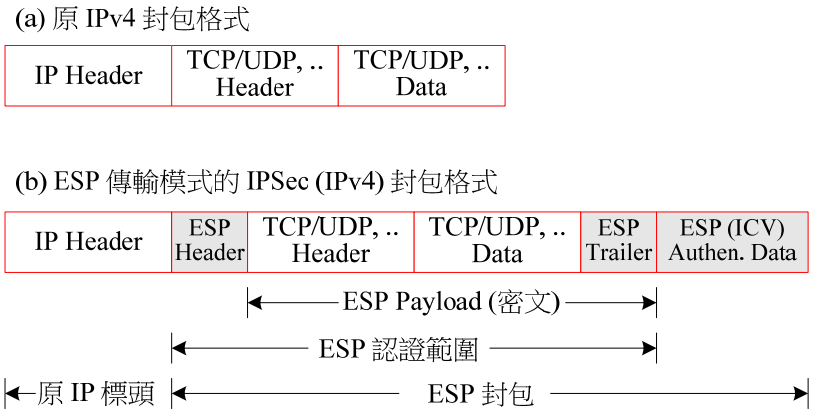


圖 10-14 IPv4 ESP 傳輸模式之封包格式

（B）IPv4 ESP 通道模式

之前我們利用圖 10-14 說明傳輸模式與通道模式之間的不同點，其中表示通道模式可使用於 NAT 網路環境裡，方法是將內部網路位址包裝在『內部標頭』（或稱原 IP 標頭）之內隱藏起來，再將『外部標頭』（或稱新的 IP 標頭）設定為合法網路位址。也就是說，IPSec ESP 封包封裝時，是將原來 IP 標頭包含進去（傳輸模式沒有），並且另外

建立一個新的 IP 標頭（外部標頭）。圖 10-15 為 IPv4 封包經由 IPSec ESP 通道模式封裝的格式，加密編碼與認證範圍如同傳輸模式一樣（ESP 封包範圍如同圖 10-16、17 一樣）。

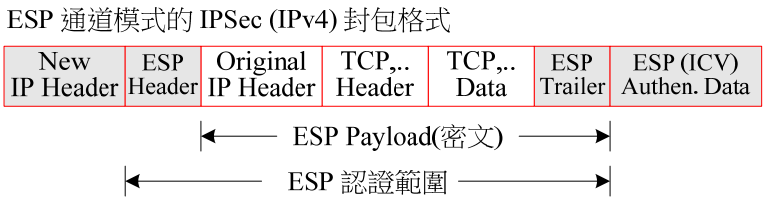


圖 10-15 IPv4 IPSec ESP 通道模式封包

10-5 ISAKMP 協定

10-5-1 ISAKMP 協定簡介

『網際網路安全關聯金鑰管理協定』（Internet Security Association Key Management Protocol, ISAKMP）是美國 NSA（National Security Agency）所制定的標準規範，並詳列於 RFC 2408。ISAKMP 是 IPSec 協定（VPN 裝置）中相當重要的協定，主要功能是建立、修改與刪除**『安全關聯』（Security Association, SA）**。其中包含協議雙方的加密金鑰、認證金鑰、以及各種演算法，也就是說，IPSec 協定所欲採用的安全措施皆是利用 ISAKMP 協定建立而成。為了因應 Internet 網路的成長，以及不同應用環境需求，雖然 ISAKMP 只提供一個安全性的基礎架構，有關較詳細的金鑰交換協定，則由另一個協定標準來規劃（IKE 協定，RFC 2409）。

ISAKMP 必須利用 UDP 協定傳輸，且由 IANA 指定在第 500 埠口（500/udp），此外，在 ISAKMP 協定上所承載訊息的編碼，全部規範於 RFC 2407，稱之為**『IPSec 解譯領域』（IP Security Domain of Interpretation, IPSec DOI）**，所以各廠商在實作 IPSec 裝置時，必

須依照 IPSec DOI 上的編碼格式。

配合前一章的敘述，我們可以瞭解建構 VPN 網路需包含 IP Sec AH、IPSec ESP、ISAKMP、IKE 與 IPSec DOI 等五個主要通訊協定，的確讓人眼花撩亂。讀者可以先參考圖 10-16 的說明，或許對瞭解這些協定之間的關係，比較容易進入狀況。說明如下：

- 倘若工作站 A 欲透過安全連線與工作站 B 通訊，兩工作站同屬一個 VPN 網路之內，但地理位置不相同，通訊連線必須經由公眾網路連接。
- 工作站 A 將封包傳送給該區域網路的安全閘門（SA_A），SG_A 判斷必須建立 IPSec 安全連線後，接著查詢本身是否有描述相關安全機制的 SA，如果有則發出 IPSec 封包（ESP 或 AH 協定）；如果沒有或 SA 過期的話，則啟動 ISAKMP 協定。
- SG_A 啟動 ISAKMP 協定與 SG_B 建立連線（500/udp），雙方協議建立 SA。其中包含協議安全套件（加密與認證系統）、安全協定（AH 或 ESP）、加密金鑰與認證金鑰等等。其實，ISAKMP 協定只提供基本架構，至於如何協議安全機制是由另一個 IKE 協定來完成。
- IKE 協定並無法獨立運作，也沒有專屬通訊的傳輸埠口，而是被嵌入於 ISAKMP 協定上，因此 IKE 協定是在 ISAKMP 的基本架構上實現，主要的功能是協議相關的安全措施。
- 建立 SA 時，必須先確認對方的身份。在安全套件裡可以指明欲利用數位憑證（PKI 或 Kerberos 系統）或帳戶/密碼（Kerberos 系統）方式，所以 ISAKMP 協定還必

須實作認證機制。

◈ 當雙方利用 ISAKMP 協定通訊時，各種協議事項都是利用 IPSec ODI 規範來編碼，只要各家廠商製造安全閘門時，都是利用 IPSec ODI 編碼各種訊息(或安全參數)，則他們之間通訊就不會發生不一致的現象

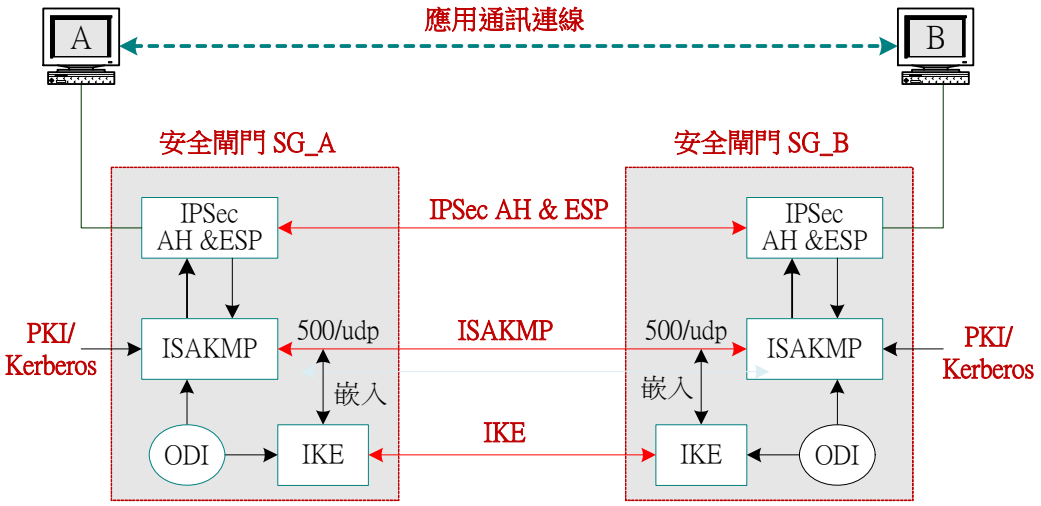


圖 10-16 IPSec 相關安全協定

簡而言之，ISAKMP 主要提供 SA 訊息的管理，也就是如何將 SA 記錄登錄於**『安全關聯資料庫』(SA Database, SAD)** 內，其中包含刪除、更新或增加 SA 記錄，並協議欲採用的安全套件。對於初學者而言，ISAKMP 與 IKE 協定很容易搞混，不易分辨兩者之間運作程序的關聯性。作者建議在研習 ISAKMP 協定時，先假設所有協議事項都是由 ISAKMP 協定所完成，暫時不要去考慮 IKE 協定；等到研習 IKE 協定時，再去考慮如何將它嵌入 ISAKMP 協定即可。

10-5-2 ISAKMP 協定功能

接下來，介紹幾點 ISAKMP 協定特性，這對我們了解 ISAKMP 協定非常有幫助。

（A）協議事項

ISAKMP 協議 SA 所需的：**安全協定（AH 或 ESP）、操作模式（傳輸或通道）、SA 壽命、認證金鑰、加密金鑰、以及各種編碼演算法**，協議後的訊息將登錄於 SAD 資料庫內，作為 IPSec 通訊時引用的 SA 安全參數。ISAKMP 並不規範鑰匙交換的運作程序，僅提出一個基本架構，此架構允許各種金鑰交換協定嵌入，譬如 IKE 協定。

（B）建立與管理 SA

若一個 SA 描述兩個或兩個以上的實體之間的安全政策，這個關係是由一組訊息來表示兩個實體之間的連接，且這些訊息必須允許實體之間的所有連線所共享。由此可見，建立 SA 完全著重於訊息的安全機制，並非僅是兩個實際連接的硬體介面。所以建立一個安全關聯是表示一個序列的安全參數，可以用一個**『安全參數索引』（Security Parameter Index, SPI）**來代表。當安全閘門（或安全主機）欲發出安全連線時，再搜尋適合的安全關聯來引用。

當 SA 不存在時（或過時失效），安全閘門（或安全主機）則需啟動 ISAKMP 協定來建立新的 SA。在 ISAKMP 協商過程中或許會啟動到其他通訊協定，如 IKE 協定，因此可能需要多個協定共同來完成。為了實現多個協定的運作，ISAKMP 協定制定一個**『起始協定交換』（Initial Protocol Exchange, IPE）**的運作程序（或稱為第一階段協商，**Phase 1 Negotiation）**。當雙方開始協議時，發起者發送 IPE 訊息給對方，其中包含一些較常用的安全套件，如果對方同意接受的話，則緊接著啟動其他協定（如 IKE）來協議該套件的安全參數；即使對方不同意，仍必須回答可以接受的安全套件。至於其他安全協定（如 IKE）可被嵌入 ISAKMP 協定之中。

（C）認證機制

為了確認建立 SA 的對方身份，ISAKMP 必須實作身份認證機制。隨著 VPN 網路的應用範圍，使用者身份認證方式也有所不同，基本上存在下列兩種方式，而且 ISAKMP 必須同時提供這兩種認證機制：

- **憑證授權(Certificate Authorities, CA)**: 使用者或組織單位由認證中心(CA Center) 發給予『數位憑證』(Digital Certificate)，通訊雙方就利用此憑證彼此認證對方身份。採用數位憑證需涉及認證公開金鑰問題，這方面請參考第九章的 PKI 系統。

- **金鑰分配中心 (Key Distribution Center, KDC)**: KDC 發給一個秘密金鑰或數位憑證給使用者(或組織單位)，通訊雙方就利用它來認證彼此身份，如 Kerberos 系統。KDC 系統較適合於企業內網路運作，也是目前許多 VPN 網路皆採用的機制。

（D）隱密性機制

為了達到建立 SA 的隱密性，ISAKMP 協定除了提供**公鑰認證機制之外，還需利用公鑰系統來交換鑰匙材料，並建立通訊所需的會議金鑰 (Session Key)**; 在通訊當中，雙方係利用秘密金鑰系統（使用會議金鑰）達到隱密性的功能。然而 ISAKMP 協定沒有定義金鑰交換協定與會議金鑰所產生的機制，這方面可透過其它協定來完成（如 IKE 協定）。

10-5-3 ISAKMP 協定堆疊

(A) 協定堆疊

圖 10-17 為 ISAKMP 協定在 Internet 網路上的協定堆疊，其中只是希望在不影響原來應用程式的架構下，將原來不可靠的 IP 協定轉換成可靠的 IPSec，換句話說，將 IP 協定轉

換成 IPSec 協定的同時，對原來 TCP 或 UDP 層次並不影響，所以原來架設在 TCP/UDP 上的應用程式，無需任何的修改也能利用 IPSec 協定來提高其安全性。其做法是先利用 ISAKMP 協定來協商(或建立 SA)，再決定何種應用程式需將 IP 協定轉換成 IPSec 協定。然而 ISAKMP 協商連線與其它應用程式一樣，都是透過 Socket 端點連線來建立，至於 ISAKMP 通訊若需要保護，可能會建立 ISAKMP SA 的安全機制，也可能透過 IPSec 協定來傳送，這完全依照雙方協商的**『保護套件』(Protection Suite)**而定。

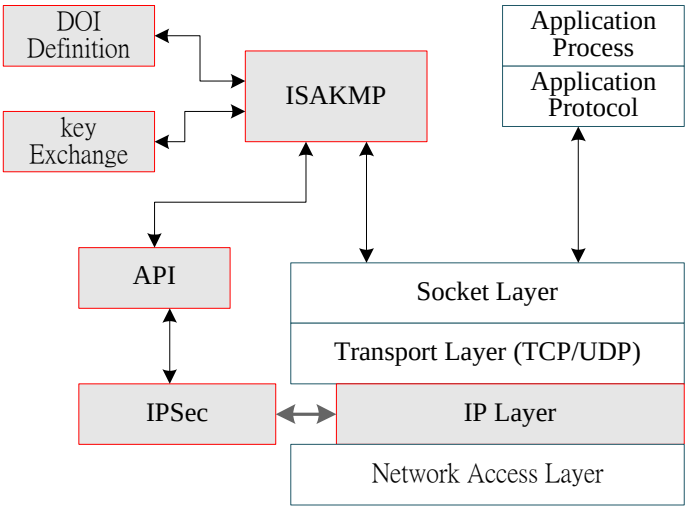


圖 10-17 IPSec 相關安全協定

(B) ISAKMP 封包標頭

ISAKMP 封包是利用 UDP 協定來傳送，並且 IANA 將 ISAKMP 固定於埠口 500 (500/udp)。圖 10-18 為 IP 的 ISAKMP 封包包裝，它除了一個封包標頭外，還包含若干筆 ISAKMP 承載 (ISAKMP Payload)，其中每一筆 ISAKMP 承載都由承載標頭 (Payload Header) 記錄所承載的訊息型態與承載資料 (Payload Data) 所組成。

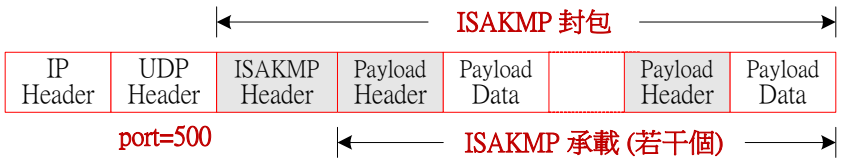


圖 10-18 IP 的 ISAKMP 封包包裝

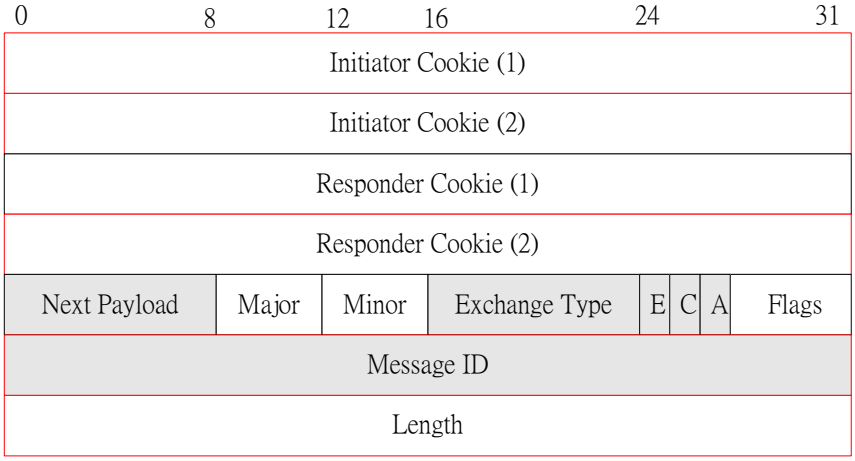


圖 10-19 ISAKMP 封包標頭格式

圖 10-19 為 ISAKMP 封包標頭的格式，各欄位功能如下：

◈ **發起者 Cookie (Initiator Cookie)**: 此欄位為 64 位元長度，是一個序號計數。表示建立 SA、通知 SA、或是刪除 SA 的發起者，以序號表示是為了防止重複攻擊及阻斷攻擊 (容後說明)。

◈ **回應者 Cookie (Responder Cookie)**: 此欄位為 64 位元長度，用來表示針對哪一個發起者 Cookie 所做的回應，存放的值就是所欲回應發起者 Cookie 的值。

◈ **下一個承載 (Next Payload)**: 此欄位為 8 位元長度，用來表示緊接著在標頭後面的第一個承載的型態。表 16-1 是 ISAKMP 目前所指定的承載型態，與其相對應的數值。

表 10-1 ISAKMP 承載封包的類型

承載型態	表示值

未用（None）	0
安全關聯承載（Security Association, SA）	1
提案承載（Proposal, P）	2
轉換承載（Transform, T）	3
金鑰交換承載（Key Exchange, KE）	4
身份標示承載（Identification, ID）	5
認證承載（Certification, CERT）	6
認證要求承載（Certification Request, CR）	7
雜湊值承載（Hash, HASH）	8
簽章承載（Signature, SIG）	9
臨時亂數承載（Nonce）	10
通知承載（Notification, N）	11
刪除承載（Delete, D）	12
製造商標示承載（Vender ID, VID）	13
保留未用	14 - 127
私人使用承載（Private USE）	128 - 255

◈ **主要版本(Master Version):** 此欄位為 4 位元長度，表示該 ISAKMP 封包的版本。在 ISAKMP 規範中規定，ISAKMP 裝置並不接受比自己版本較高的封包。

◈ **次要版本（Minor Version）**：此欄位為 4 位元長度，表示該 ISAKMP 封包的次要版本。

◈ **交換種類（Exchange Type）:** 此欄位為 8 位元長度，表示此封包訊息的交換型態。

表 10-2 為 ISAKMP 協定所定義的交換型態：

表 10-2 交換類別

交換型態	表示值
未使用（None）	0
基本交換（Base Exchange）	1
身份保護交換（Identity Protection Exchange）	2
僅認證交換（Authentication Only Exchange）	3
積極交換（Aggressive Exchange）	4
訊息交換（Information Exchange）	5
ISAKMP 未來使用	6-31
DOI 描述使用	32-239
私人使用	240-255

◈ **旗標(Flag):** 此欄位為 8 位元長度，目前只用到 3 個位元，其他位元都先設定為 0。

旗標位元表示封包的狀態，說明如下：

1. **編碼位元（Encryption bit, E）:** 此位元被設定時，表示標頭之後的承載都已被 ISAKMP 所指定的加密演算法編碼，否則表示承載資料都是明文。

2. **承諾位元(Commit bit, C):** 此位元是使用於單一鑰匙交換同步，以確保在 SA 建立完成之前不會收到任何加密的資料。此位元設定為 1 乃表示雙方已協議

出秘密金鑰，若其中一方設定為 1 時，就必須等到另一方同樣設定為 1，方可認定雙方都已確定秘密金鑰，始可傳送已加密的訊息。

3. **僅供認證位元（Authentication Only Bit, A）**：此位元設定為 1 時，表示此封包僅採用 SA 所指定的認證演算法進行完整性檢查，但不採用加密方法來保護訊息。

◈ **訊息標示（Message ID）**：此欄位為 4 個位元長度，包含由發起者在第二階段協議中所產生的隨機數字。如果有兩個以上的 SA 建立連線傳送訊息（如碰撞現象），此訊息標示可以顯示出之間的不同，而被用來識別第二階段連線的唯一表示值。

◈ **長度（length）**：此為 4 個位元長度的欄位，記錄整個封包的長度，其中包含封包標頭與承載資料，以位元組為計算單位。

『cookie』是一個 ISAKMP SA 通訊連線的標示，其功能和 IPSec 協定（AH 或 ESP）封包標頭上的序號（Sequence Number）欄位非常相似，除了當作連線識別之外，還必須具有防止反重播攻擊與阻斷攻擊。為了預防攻擊者猜測出，某一個 ISAKMP SA 連線是屬於那一個安全閘門所建立的，cookie 大多是利用雜湊演算法，計算出產生者（發起者或回應者）所欲協調安全機制的 IP 位址與傳輸埠口，並加入日期與時間戳記。

10-6 IKE 協定

10-6-1 IKE 協定簡介

(A) IKE 協定功能

『網際網路金鑰交換』（Internet Key Exchange, IKE）是一種混合協定，係將

Oakely 與 SKEME 兩個鑰匙交換協定結合於 ISAKMP 協定上，標準規範由 RFC 2409 文件敘述。ISAKMP 協定為了提高鑰匙交換的變異性，僅制定鑰匙交換的基礎架構，並沒有規範鑰匙交換的實作規範，一方面為了配合鑰匙技術的快速發展，另一方面也可阻斷推陳出新的駭客攻擊手段。倘若 ISAKMP 同時制定了鑰匙交換的實作協定，一旦有更新的隱密技術或攻擊手法被發展出來時，勢必需修改其運作程序，才能滿足新的環境需求。值得注意的是，一個已被 Internet 網路廣泛使用的通訊協定，無論是修改或增加功能，都是耗時費力的重大工程。譬如，已在網路上應用一段不短時日的 IPv4，很多人都知道若想把它改成 IPv6，實在是困難重重。ISAKMP 協定有鑑於此，於是將鑰匙交換協定分離出來，並以選項的方式，由各組織單位依其所需選擇不同的交換協定，不但可以滿足不同環境的需求，也可隨時增加它的功能。雖然目前 IKE 只制定 Oakely（RFC 2412）與 SKME [91] 兩種交換協定，但相信陸續會有更新的協定被加入。

到底 IKE 協定是如何將 Oakely 與 SKEME 協定嵌入 ISAKMP 協定中？記得我們在介紹 ISAKMP 協定時，曾經提到許多與鑰匙或認證有關的承載，譬如，金鑰交換承載、雜湊承載、簽章承載等等，其承載資料中並沒有指定何種資料型態，大多只說明依照 ODI 編碼方式。又許多交換型態，譬如，部份保護交換、僅供認證交換等等，也沒有說明如何產生加密或認證金鑰。既然 ISAKMP 協定僅提供一個運作架構，IKE 協定就是在這運作架構中來實作鑰匙交換的工作，簡單的說，IKE 就是利用 ISAKMP SA 中，與鑰匙交換有關的承載或交換型態，作為實作鑰匙交換的工作；然而協議雙方之間鑰匙交換的運作程序就依照 Oakely 與 SKEME 協定所規範的方式。礙於篇幅，本書主要以 Oakely 協定為介紹重點（RFC 2409 也是如此）。(請參閱『**資訊與網路安

全技術』)

(A) IKE 協定特性

IKE 協定主要是提供 IPSec 通訊中所需的一切『**鑰匙**』(**Key，或稱金鑰**)，其中除了包含加密與認證金鑰的產生，還可選擇不同的鑰匙產生技術、以及協議各種加密或認證系統。我們首先將 IKE 協定的一些特性歸類起來，希望對讀者有所幫助：

- **兩階段協商**：為了配合 ISAKMP 協定運作，IKE 亦採用兩階段的協商方式，第一階段協商是建立安全通訊連線；第二階段才真正進入鑰匙交換程序。
- **協商項目**：IKE 協定至少必須協商下列相關演算法：
  - **加密演算法**(**Encryption Algorithm**)，如 DES。
  - **雜湊演算法**(**Hash Algorithm**)，如 MD5 或 SHA。
  - **認證方法**(**Authentication Method**)，如 HMAC。
  - **Diffie-Hellman 演算法**所需的訊息群組，如 MOPD(容後介紹)。
- **認證或加密金鑰產生**：在 IKE 協定上，針對認證或加密金鑰產生有下列三種方式：
  - **預先共享金鑰**(**Pre-Shared Key**)：雙方通訊之前利用其他管道，將秘密金鑰分配給雙方；一般大多採用 KDC 系統(如 Kerberos)系統來分配金鑰。
  - **會議金鑰**(**Session Key**)：雙方利用 Diffie-Hellman 演算法，以互相交換鑰匙材料，所計算產生的金鑰。

- **公開金鑰（Public key）**：係利用憑證授權（CA）中心所發給的公鑰，互相認證彼此身份，並利用它傳遞**『主密鑰』(Master Secret)**。

◈ **『完全順向密鑰』(Perfect Forward Secret, PFS)**：如果僅能用一把秘密金鑰加密或解密，而沒有另外其他鑰匙可以取代的話，稱之為『完全順向加密』（PFS）。IKE 協定所建立的會議金鑰，就是必須達到 PFS 的特性。

◈ **『虛擬亂數函數』(Pseudo-Random Function, PRF)**：係嵌入鑰匙的雜湊演算法，如 HMAC-MD5 等。IKE 為了增加鑰匙的複雜度，通常利用 PRF 函數計算鑰匙材料，以得到相關鑰匙參數。但許多金鑰交換協定之中，並未包含協議 PRF 函數的產生方式，因此，實作時必須規範採用何種 PFR 函數。

其實計算鑰匙時，如何使所產生的鑰匙具有完全順向密鑰（PFS）功能，需賴 PRF 函數的重複計算，唯經由 PRF 函數重複計算所產生的鑰匙，比較能接近 PFS 所要求的功能。

10-6-2 IKE 交換模式

ISAKMP 主要定義金鑰交換與各種承載（ISAKMP Payload）的基礎架構，而 IKE 協定則將實現的金鑰交換程序嵌入 ISAKMP 所定義的架構，即 IKE 協定（Oakely 與 SKME）係利用 ISAKMP 承載作為攜帶交換鑰匙的訊息，並且依照 ISAKMP 交換類別（16-8 節介紹）達成金鑰交換的運作程序。換句話說，Oakely 主要定義運作『模式』（Mode），而 ISAKMP 則定義協商**『階段』(Phase)**，至於如何將『模式』的運作程序嵌入協商『階段』裡，則與交換功能的型態有關。

(A) 兩階段協商模式

簡單的說，就 ISAKMP 的基礎架構而言，第一階段協商既然是建立安全性的 ISAKMP SA，所以 Oakely 必須將某些運作『模式』嵌入此階段之中；同樣的道理，第二階段既然 ISAKMP 利用所建立的安全連線協商相關 SA 的參數值，所以 Oakely 亦需將某些『模式』嵌入其中。至於兩個階段中應該嵌入那些『模式』，則必須視通訊所需的交換型態而定，因此我們必須先了解 Oakley 到底定義了那些『運作模式』才行，以下有四種定義：

- **主要模式（Main Mode）**：主要功能是協商雙方安全機制，協議出認證與加密所需的會議金鑰（也許使用 Diffie-Hellman 演算法），並認證雙方身份。其實它的功能非常類似於 ISAKMP 的『識別保護交換』；此模式主要應用於 ISAKMP 的第一階段協商。

- **積極模式（Aggressive Mode）**：此模式非常類似於 ISAKMP 的『積極交換』，其實就是它的實作。積極模式是為了簡化主要模式而來，主要目的是要認證雙方身份，因此可以應用於 ISAKMP 協定的第一階段協商。

- **快速模式（Quick Mode）**：此模式主要應用於有保護的連線下，從事鑰匙交換的工作，所以在此模式下交換的承載，都經過加密編碼處理過；此模式主要應用於 ISAKMP 的第二階段協商（與任何 ISAKMP 交換不類似）。

- **新群組模式（New Group Mode）**：主要應用於 Diffie-Hellman 演算法協議出新的演算群組。在 IKE 協定上，Diffie-Hellman 有三個亂數群組供建立會議

金鑰所需，且這些群組是公開的，此運作模式就是用來協商使用哪一個群組；

此模式可應用於 ISAKMP 的兩個協議階段（與任何 ISAKMP 交換不類似）。

由以上敘述可以了解，ISAKMP 的第一階段協商中可能用到主要模式、積極模式或新群組模式，至於第二階段協商則可能用到快速模式或新群組模式。無論何種模式都必須被嵌入於 ISAKMP 交換種類之中，其中主要模式與積極模式可能被嵌入的交換類別如表 10-2 所示。但快速模式與新群組模式並不屬於 ISAKMP 交換種類，因此，IANA 另外定義兩個交換種類以滿足 Oakely 協定的需求，表示值如下：

- 快速模式：32
- 新群組模式：33

(B) 承載訊息表示法

無論是主要模式、積極模式、快速模式或新群組模式，都是由 ISAKMP 的各種承載來封裝。譬如，主要模式或積極模式是利用提案承載來封裝，而每個提案承載中可再包含若干個轉換承載，其中每一轉換承載表示一個安全機制的訊息。在進入運作程序前，我們先介紹會使用到的承載與訊息之表示方法：

- **HDR**：代表 ISAKMP 標頭（Header）；而 HDR* 表示標頭以後的承載都是經過加密編碼的。
- **SA**：代表一個 SA 承載，其中可能攜帶一個或一個以上的提案承載，以及其相關的轉換承載。
- **KE**：代表金鑰交換承載。

- **IDx：**代表身分證明承載，若 x = ii 或 ir，係表示 ISAKMP 發起者(Initiator)與回應者(Responder)的身份證明，若 x=ui 或 ur，則表示使用端(非 ISAKMP SA) 之發起者與回應者的身分證明。
- **<P>_b：**ISAKMP 封包內的承載實體 (Payload Body)，其中不包含承載標頭 (Payload Header)。
- **HASH：**雜湊承載。
- **CERT：**憑證承載。
- **SIG：**簽章承載。
- **Nx：**隨機承載(Nonce Payload)，若 x = i 表示此隨機值是由發起者(Initiator) 傳送若 x = r，則表示由回應者 (Responder) 發出。
- **Ck-I 與 Ck-R**：ISAKMP 封包標頭上的發起者與回應者的 Cookie 值。
- **g^{xi} 與 g^{xr}：**發起者與回應者的公開 Diffie-Hellman 值。
- **prf(key, msg)：**表示利用鑰匙 key 對訊息 msg，經過某一種雜湊演算法計算後，所得到『虛擬亂數函數』(PRF) 的值。
- **SKEYID：『共享金鑰標示』**；係由雙方交換**『秘密材料』(Secret Material)** 所推演出來的一個字串，並作為產生各種鑰匙的依據。
- **SKEYID_e**：係 ISAKMP SA 為了達成資料隱密性功能，所需產生加密金鑰的**『鑰匙材料』(Keying Material)**；由 SKEYID 計算而來。
- **SKEYID_a**：係 ISAKMP SA 為了達成認證功能，所需產生認證金鑰的『鑰匙材料』；由 SKEYID 計算而來。
- **SKEYID_d**：是非 ISAKMP SA 所需產生『衍生鑰匙』(Derive Key) 的『鑰匙

材料』；由 SKEYID 計算而來。

- **<x>y**：代表資料（x）經過鑰匙（y）所編碼。
- **x | y**：代表 x 和 y 串接（Concatenation）在一起。
- **[x]**：代表 x 是選項，可有可無。

10-6-3 IKE 鑰匙計算

『共享金鑰標示』（Shared Key Identifier, SKEYID）係計算各種鑰匙材料的基本要素，也是 IKE 協定制定 SKEYID 的基本產生方法。在第一階段協商中包含：數位簽章認證、公鑰認證、修正型公鑰認證、以及預先共享金鑰認證等四種方法，每一種認證系統所產生 SKEYID 的方法如下：

數位簽章：SKEYID = prf(Ni_b | Nr_b, g^{xy})

公鑰加密：SKEYID = prf(hash(Ni_b | Nr_b), CKY-I | CKY-R)

共享金鑰：SKEYID = prf(pre-shared-key, Ni_b | Nr_b)

其中 prf 表示採用某一種 PRF 函數。基本上，IKE 並沒有規定應該採用何種演算法，但至少必須提供 HMAC-MD5 與 HMAC-SHA 兩種方法。

經過主要模式或積極模式交換訊息之後，可計算出下列三種鑰匙材料：

SKEYID_d = prf(SKEYID, g^{xy} | CKY-I | CKY-R | 0)
SKEYID_a = prf(SKEYID, SKEYID_d | g^{xy} | CKY-I | CKY-R | 1)
SKEYID_e = prf(SKEYID, SKEYID_a | g^{xy} | CKY-I | CKY-R | 2)

其中 “prf” 是雙方協議的某一種雜湊演算法計算出來的虛擬亂數；g^{xy} 是雙方互相傳送

鑰匙材料之後，並經由 Diffie-Hellman 演算法計算得來的共享密鑰（Shared Secret）。CKY-I 與 CKY-R 是由 ISAKMP 封包標頭上取得的發起者與回應者 Cookie。為了提供未來通訊安全機制的延伸，數值 0、1、2 表示一個位元組的空間，以作為未來擴充使用（請參考 RFC 2409）。

依照 IKE 協定，在認證過程的交換訊息當中，發起者需產生 HASH_I，回應者產生 HASH_R 並作為雙方認證使用，產生的方法如下：

$$HASH\_I = prf(SKYID, g^{xi} \mid g^{xr} \mid CKY\text{-}I \mid CKY\text{-}R \mid SAi\_b \mid IDii\_b)$$

$$HASH\_R = prf(SKYID, g^{xr} \mid g^{xi} \mid CKY\text{-}R \mid CKY\text{-}I \mid SAi\_b \mid IDir\_b)$$

其中 g^{xi} 與 g^{xr} 表示雙方所交換的鑰匙材料、SAi_b 表示發起者 SA 承載實體（Payload Body）、IDii_b 與 IDir_b 表示發起者與回應者的身份識別承載實體。HSAH_I 與 HASH_R 主要運用在數位簽章認證方面，作為簽章與身份證明時使用；在公開金鑰或共享金鑰認證方面，則作為交換認證使用。

10-6-4 IKE 第一階段協議

第一階段協商，雙方除了協議安全機制外，還必須互相確認對方身份。基本上，協議安全機制的方法大多與認證方法有關，在 IKE 協定上，第一階段有下列四種身份認證方法：簽章認證、公開金鑰認證、修正型公鑰認證、預定共享金鑰認證，以下分別介紹這四種認證方法。

(A) 使用簽章認證

使用簽章認證表示雙方利用一個『**共享密鑰**』**(Shared Secret)**向某一筆資料簽署，再互相傳送給對方作為認證身份時使用。由此可見，雙方必須利用 KE 承載交換鑰匙

材料，以建立雙方共享密鑰（SKEYID_a），並交換亂數承載（Nonce, Nr, Ni）來建立雜湊亂數（HASH_I 與 HSAH_R）。簽章認證有主要模式與積極模式，如圖 10-20 所示，主要模式的運作程序說明如下：

- **編號 (1) 與 (2)**：發起者與回應者之間協調安全機制（ISAKMP SA），譬如，協議雙方進行鑰匙交換時，所採用的是 AH-MD5 安全套件（其他套件如表 16-4 所示）。

- **編號 (3) 與 (4)**：雙方進行鑰匙交換程序，其中 KE 承載為雙方進行 Diffie-Hellman 演算法所傳送的鑰匙材料。譬如，發送端傳送 g^x 給回應端，且回應端亦傳送 g^y 給發起者，雙方利用此鑰匙材料建立會議金鑰為 g^{xy}。亂數承載（Ni 與 Nr）是作為反重複攻擊與計算 HASH_I 和 HSAH_R 時使用。

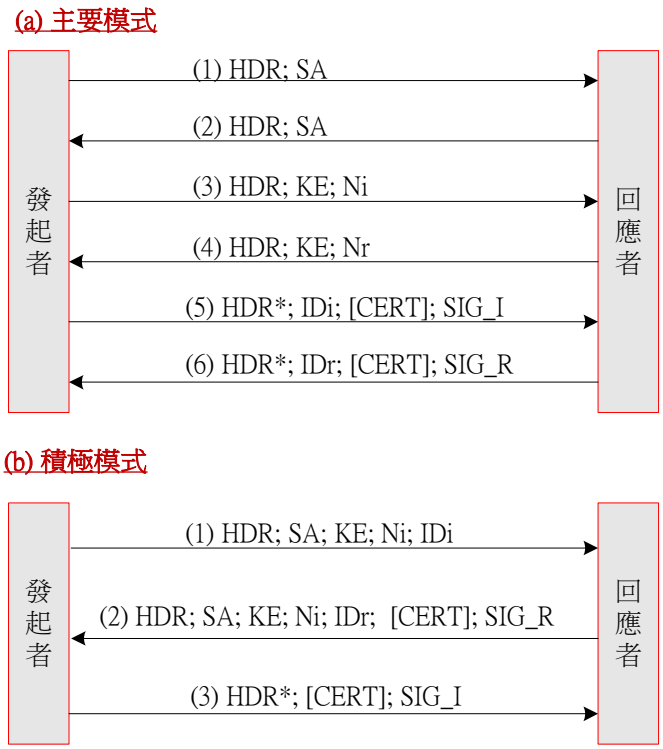


圖 10-20 IKE 第一階段 – 使用簽章認證

◈ **編號 (5) 與 (6)**：雙方除了利用會議金鑰將身分證明（或身份憑證）加密外，同時與協議出的數位簽章演算法（如 HMAC-MD5)計算出虛擬亂數函數（PRF），並以 SIG_I（或 SIG_R）承載傳送給對方。

簡單的說，訊號 (1) 與 (2) 是雙方協議的安全套件，訊號 (3) 與 (4) 是交換所協議安全套件的相關鑰匙材料，再由這些材料計算出共享密鑰（SKEYID）與雜湊值，計算方式如下：(與前面介紹的相同)

$$SKEYID = prf(Ni\_b \mid Nr\_b, g^{xy})$$
$$SKEYID\_d = prf(SKEYID, g^{xy} \mid CKY\text{-}I \mid CKY\text{-}R \mid 0)$$
$$SKEYID\_a = prf(SKEYID, SKEYID\_d \mid g^{xy} \mid CKY\text{-}I \mid CKY\text{-}R \mid 1)$$
$$SKEYID\_e = prf(SKEYID, SKEYID\_a \mid g^{xy} \mid CKY\text{-}I \mid CKY\text{-}R \mid 2)$$
$$HASH\_I = prf(SKYID, g^{xi} \mid g^{xr} \mid CKY\text{-}I \mid CKY\text{-}R \mid SAi\_b \mid IDii\_b)$$
$$HASH\_R = prf(SKYID, g^{xr} \mid g^{xi} \mid CKY\text{-}R \mid CKY\text{-}I \mid SAi\_b \mid IDir\_b)$$

接下來，訊號 (5) 與 (6) 係利用上述所建立的共享密鑰來交換訊息，譬如利用 SKEYID_a 簽署 HASH_I 與 HASH_R 以得到 SIG_I 與 SIG_R；或利用 SKEYID_e 向所承載的資料加密。另外，憑證承載（CERT）為選項項目，如沒有的話，則需藉身份識別承載來互相確認身份。

圖 10-20 (b) 為積極模式的運作程序，可以看出他是由主要模式（圖 10-20 (a)）簡化而來，簡化的方式非常類似 ISAKMP 的積極交換。雖然積極模式對於雙方身分識別承載（IDi 與 IDr）並沒有經過加密保護，但在產生 HASH_I 與 HASH_R 時已將這兩個承載加入，同時已得到相當的保護作用；所以攻擊者竄改 ID 承載時，雙方亦可由 SIG_I 與 SIG_R 檢測出來。另外，沒有利用憑證驗證身份，很容易遭受中間人攻擊，此點於應用時必須特別注意。因為積極模式的運作程序與主要模式大同小異，這裡就不

再贅言了。

(B) 使用公鑰認證

利用**公鑰（Public Key）認證**，雙方雖無需協議出會議金鑰，但還是需要一個憑證授權（CA）中心來發行雙方公鑰（如 PKI 系統）。它的運作程序是雙方交換訊息（身份識別或亂數）時，傳送者先利用對方的公鑰加密，對方收到後再利用自己的私鑰（Private Key）解密，如此便能達到認證性與隱密性的功能。採用公鑰認證，必須使用非對稱加密演算法(如 RSA 演算法)，IKE 協定建議至少必須提供 PKCS #1 密碼系統。圖 10-21 為使用公鑰認證的運作程序，同樣可利用主要模式與積極模式來實現，我們就主要模式來介紹，有關積極模式的運作既然由主要模式簡化而來，在此不再另述（圖 10-21 (b))。圖 10-21 (a)是以公鑰認證建立第一階段的協商，說明如下：

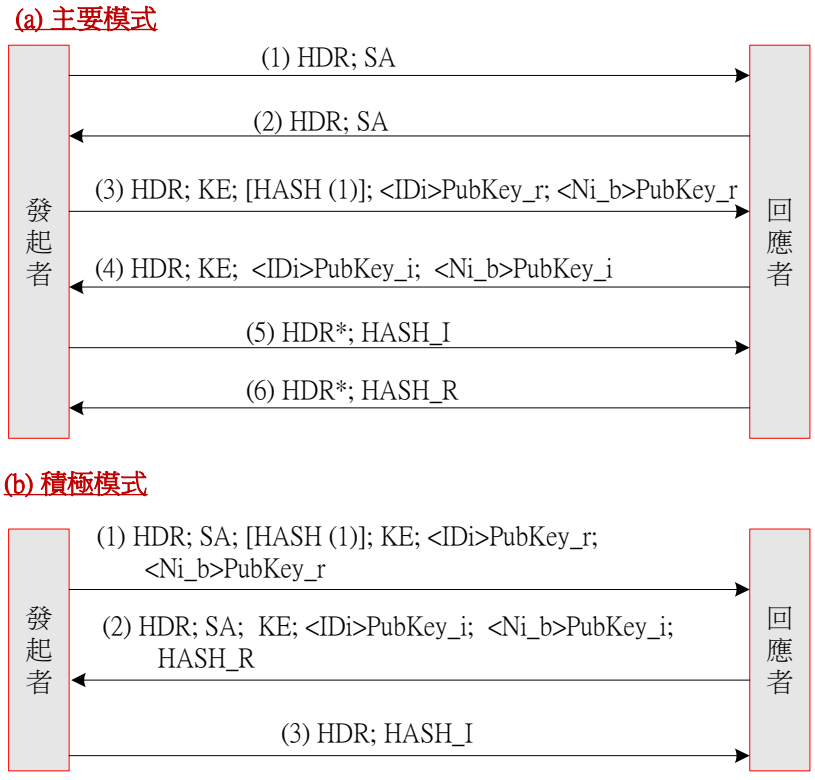


圖 10-21 使用公鑰認證之運作程序

◈ **編號 (1) 與 (2)**：雙方以 ISAKMP SA 協議採用何種安全機制。此範例是協議以公鑰認證機制，並保護雙方傳送的訊息。

◈ **編號 (3) 與 (4)**：雙方互相以對方的公鑰，向自己的身分識別承載加密（<IDi_b>PubKey_r 或 <IDr_b>PubKey_i），對方再以自己的私鑰(Private Key)解密。如果回應者擁有許多不同身份（或擁有許多 CA 中心），而有多把公鑰時，發起者必須加入亂數並將其加密（<Ni_b>PubKey_r），而身分證明與亂數的加密鑰必須相同。HASH(1) 是亂數與身分證明的雜湊值(雙方已協議出雜湊演算法)，回應者利用私鑰向身分證明與亂數解密後，再依照協議的亂數演算法，計算出亂數值，如果與 HASH(1) 相同的話，表示使用相同配對的公/私鑰。除了雙方使用公鑰認證外，也可以傳送鑰匙材料（KE）來協議雙方的會議金鑰（Diffie-Hellman 演算法），作為爾後通訊時使用。

◈ **編號 (5) 與 (6)**：雙方利用公鑰（或會議金鑰）來保護協議中的訊息（隱密性功能），也可以利用雜湊演算法來認證雙方訊息的正確性（HASH_I 與 HASH_R）。

其中 HASH(1) 是一個雜湊值（雙方協議之雜湊函數），它是由發起者的亂數與身份承載所計算得來的，雖然可以避免中間人攻擊，但仍逃不過阻斷攻擊。

(B) 使用修正型公鑰認證

使用公鑰認證最大的缺點是，加密與解密的負荷過大（通常需較長的計算時間），這也是非對稱加密演算法的優點(增加安全性)。更何況公鑰認證需要四次加密/解密的過

程（兩次公鑰加密、兩次私鑰解密），如此更顯得公鑰認證的低效益。如果我們可以稍加修改，在公鑰基礎架構上協議出對應的加密金鑰，然後利用對稱加密演算法來保護雙方所傳送的訊息，如此便可以提高效益了，這就是『修正型公鑰認證』的基本構想。

它的做法是，利用雙方所交換的 SA 與 Nonce 承載，各自製造出一個加密金鑰，然後雙方再利用此鑰匙向所交換的身份識別與鑰匙材料承載加密，如此一來，除了可以減少加密與解密的負荷，還可以增加安全性。只要將製造共享密鑰的亂數（Ni 與 Nr）利用對方的公鑰加密，接收端再利用自己的私鑰解密，攻擊者就很困難得到製造共享密鑰的材料。圖 10-22 為修正型公鑰認證之主要模式和積極模式的運作模式，我們還是用主要模式來介紹它的運作程序，說明如下：

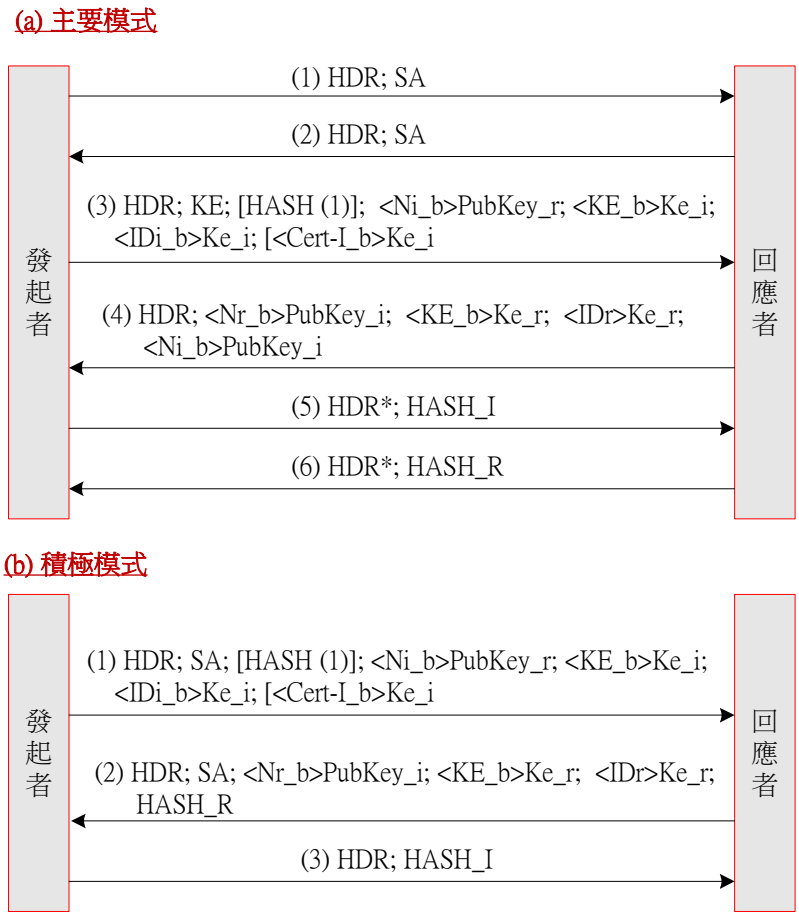


圖 10-22 修正型公鑰認證之運作程序

◈ **編號 (1) 與 (2)**：雙方利用 SA 承載協調安全機制。

◈ **編號 (3) 與 (4)**：發起者利用對方公鑰向亂數（計算共享密鑰使用）加密，並與身份證明一同計算雜湊值（HASH(1)），傳給對方確認封包未經竄改。雙方利用協議的會議金鑰（Ke_i 與 Ke_r），向鑰匙材料（KE_b）與身分證明（IDi 與 IDr）加密，並相互傳輸。任何一方除了可以認證對方身份外，並可以協議出通訊所需的共享密鑰，亦即利用雙方的鑰匙材料（KE_b）所計算出來（容後介紹）。

◈ **編號 (5) 與 (6)**：雙方利用會議金鑰來執行加密與數位簽章的功能（HASH_I 與 HSAH_R，如 10-5-3 節介紹）。

到底如何利用 SA 交換來建立雙方的認證金鑰（Ke_i 與 Ke_r，兩者不相同）？簡單的做法是，自己先製造出自己的認證金鑰（Ke_i 或 Ke_r），並對某些訊息加密（ID 與 KE），再將製造出來的材料（SA 與 Nonce）傳送給對方；對方收到這些材料後，一樣製造出它的認證金鑰，並對他所傳送的訊息解密；其中我們只要針對某些材料（Nonce），使用對方的公鑰加密，就不怕他人窺視。

在編號 (3) 與 (4) 協議中，雙方互送一個經由對方公鑰加密過的亂數承載，其中只針對承載的內容加密（<Ni_b>Pubkey_r 與 <Ni_b>PubKey_i），而就利用這個亂數與雙方的 Cookie 號碼（CKY_I 與 CKY_R）製作會議金鑰的材料（Ne_i 與 Ne_r），如下：

Ne_i = prf(Ni_b, CKY-I)

Ne_r = prf(Ni_b, CKY-R)

其中“prf”為雙方協議的雜湊演算法的『虛擬亂數函數』(PNF)。接下來，我們以發起者 (Initiator) 為範例，說明製造認證與加密金鑰 (訊號 (5) 與 (6) 使用) 的步驟，如下：

K1 = prf(Ne_i, 0)

K2 = prf(Ne_i, K1)

K3 = prf(Ne_i, K2)

則：Ke_i = K1 | K2 | K3 (”|”為串接記號)

在製造過程中，如果 K1 (或 K2、K3) 超過鑰匙所規範的長度時，只取較高位元部份，並捨棄較低位元。至於然而回應者的認證金鑰如同發起者一樣，不再另述。另外，如果採用 CBC 模式的對稱加密法時需要起始向量 (IV)，它將會被第一個 Nonce 承載所攜帶，並利用鑰匙 Ne_i 或 Ne_r 加密著，隨著此承載後面才是傳輸資料(HASH_I 或 HSAK_R)，並利用鑰匙 K 加密著。

(C) 使用預定共享鑰匙認證

『預定共享鑰匙』(Pre-Shared Key) 表示雙方為通訊之前，就各自擁有一支共享金鑰。至於此鑰匙是如何分配得來(經由人工轉送或其他等等)，不在 IKE 的處理範圍內。預定共享鑰匙確認的做法是，雙方交換鑰匙材料之後，再配合共享鑰匙製造出一把『共享密鑰』，如下：(如 10-5-3 介紹)

SKEYID = prf(pre-shared-key, Ni_b | Nr_b)

接著再利用此共享密鑰 (SKEYID) 製造出其他加密金鑰。

圖 10-23 所示為使用預定共享鑰匙認證的運作程序，其中包含主要模式(圖 10-23

(a))與積極模式(圖 10-23 (b))，我們依然只針對主要模式的運作程序加以介紹，如下：

- **編號 (1) 與 (2)**：發起者與回應者之間協調 ISAKMP SA 的安全機制。
- **編號 (3) 與 (4)**：雙方交換亂數（Ni 與 Nr），並配合共享金鑰製造出共享密鑰（SKEYID）。另外，鑰匙材料承載（KE）在 RFC 2409 內並沒有明確規範使用方法（作者也不清楚），可能是攜帶起始向量時使用。
- **編號 (5) 與 (6)**：雙方利用該共享密鑰向身分證明（IDi 與 IDr）加密，並計算數位簽章（或雜湊演算）(HASH_I 與 HASH_R)，以便互相通訊。

圖 10-23 預定共享金鑰認證之運作程序

一般積極模式主要應用於身分證明不需保護的情況下，所以通常僅做數位簽章（HASH_I 與 HASH_R），不適用於加密的動作。

10-6-5 IKE 第二階段協商

經過第一階段協商之後，通訊雙方已協議出安全套件（ISAKMP SA），並可確認對方身份無誤（交換 ID 或 CERT 承載），也已協議出共享密鑰（SKEYID），以及加密或認證密鑰（SKEYID_e、SKEYID_a 與 SKEYID_d）。如此一來，便可利用這些安全機制進行第二階段的協商。換句話說，第二階段是利用第一階段所建立的安全通道，來協商 IPSec SA 的安全機制，而且第二階段協商是希望在安全連線之下，建立相關 IPSec SA 的安全套件。我們可以發現，經過許多的運作程序，至今才真正進入協商『安全閘門』所需要的安全機制 SA，唉！可真辛苦。如此嚴密的措施就是為了防止在協商 SA 時，遭受攻擊者的竄改、偽裝或竊聽。到目前為止，總算可以清楚的分辨：

- **ISAKMP SA**：在第一階段的安全機制，並依此安全機制來保護第二階段協商。
- **IPSec SA**：在第二階段中協商 VPN 的安全機制，又依照 SA 內所指定的安全協定，可區分為 AH SA 或 ESP SA 兩種。

然而，第二階段協商又可區分為下列三種運作模式：

- 快速模式
- 新群組模式
- ISAKMP 訊息交換

以下分別介紹之。

(A) 快速模式

『快速模式』(Quick Mode) 並不是一個完整的訊息交換程序，僅是延續第一階段訊息的交換。也就是說，它必須配合第一階段，其中可能是簽章認證、公鑰認證、修正

型公鑰認證或預定共享金鑰認證的交換訊息。並且快速模式的交換過程當中，所有 ISAKMP 封包的承載都是經過加密，並且在 ISAKMP 標頭 (HDR) 後面必須緊接著一個 HASH 承載，此 HASH 承載是用來認證該訊息，以及作為存活的證明。接下來 SA 承載必須緊接在 HASH 承載之後，而在 SA 承載之後必須緊接著 Nonce 承載，Nonce 承載上的亂數是作為反重複攻擊使用。如果需要利用提供 Diffie-Hellman 演算法來協商會議金鑰時，會緊接著一個攜帶鑰匙材料的 KE 承載，如需要身份識別，則必須加入 ID 承載。在這些承載之中，必須強調下列重點：

- **HDR 標頭**：如同圖 10-24 中的連線類別 (3) 與 (4)，也就是說，該 HDR 標頭的 Message-ID 與 SPI 欄位已有固定數值。
- **SA 承載**：表示欲協商安全套件，因此，它的後面會緊隨著一些提案承載，與相對應的轉換承載。
- **ID 承載**：表示欲協商安全機制的身份識別，可能是某一網路位址(IP 位址)，或網路位址附加傳輸埠口 (TCP/UDP 埠口) (表示伺服器設備)。

第二階段協商是採用連線類別 (3) 與 (4)，表示之前雙方已協議出訊息標示 (Message-ID)與 SPI 值。然而 ISAKMP 封包也是利用這兩個欄位來識別是 ISAKMP SA 或 IPSec SA。

在快速模式下，有一個重要的安全機制，稱之為『**完全性順向機密』(Perfect Forward Secrecy, PFS)**。在 PFS 機制上，雙方協議出一把保護資料的鑰匙，就不能允許另一把鑰匙的產生；換句話說，如果某鑰匙是被用來保護傳輸資料，則製造此鑰匙的

材料，就不能拿來再製作其他鑰匙。如果依照這個機制，就不會有第二把重複鑰匙的產生；更進一步，即使此鑰匙被破解，仍不會影響到其他鑰匙所保護的資料，因為，鑰匙之間並沒有重複的資料可尋。

圖 10-24 為快速模式的運作程序，說明如下：

◈ **編號 (1) 與編號 (2)**：發起者與回應者之間互相協議安全機制，如果協議中要求使用『完全性順向機密』(PFS) 的機制時，雙方訊息必須包含鑰匙交換的訊息 (KE)。另外，整個封包除了標頭以外，都經過第一階段所協議的安全機制加密過。

◈ **編號 (3)**：此封包是確認前面的交換訊息。

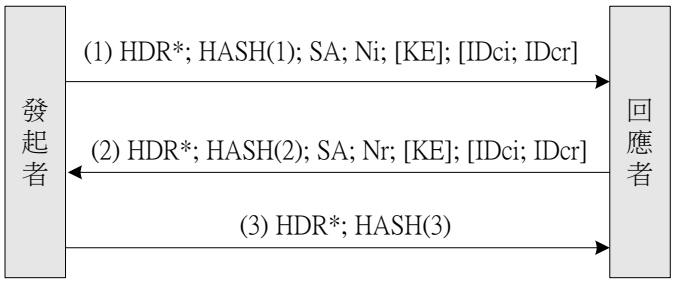


圖 10-24 快速模式的運作程序

在圖 10-24 中有三的重要的雜湊值，如下：

HASH(1) = prf(SKEYID_a, M-ID | SA | Ni [| KE] [| IDci | IDcr])

HASH(2) = prf(SKEYID_a, M-ID | Ni_b | SA | Nr | [| KE] [| IDci | IDcr])

HASH(3) = prf(SKEYID_a, 0 | M-ID | Ni_b | Nr_b)

其中 prf 表示經由協議出來的雜湊演算法，所計算出來的『虛擬雜湊函數』(PRF)，M-ID 表示 ISAKMP 標頭上訊息欄位的『訊息標示』(Message ID)(如圖 16-4 所示)。在第

二階段協商中，M-ID 是由發起者所產生的一個亂數，作為雙方連線的標記。IDci 與 IDcr 表示發起者與回應者欲協商設備的識別碼（c 表 Client 的意思），其中可能包含通訊協定、傳輸埠口、使用者名稱等等。另外，SKEYID 的產生方法請參考 16-4-3 節介紹。

依照不同的安全機制，對於 HASH 產生方式有許多選項，譬如協議時，需要檢視較詳細的雙方身份資料（如通訊協定、傳輸埠口、使用者名稱等等），則必須將 IDcr 與 IDci 兩個承載加入雜湊演算法之中。更重要的，如果雙方選擇採用『完全性順向機密』（PFS）機制時，表示雙方必須協議出一個獨一無二的會議金鑰，因此便需交換鑰匙材料（KE），也會將他加入 HASH 計算之中。

接下來，探討如何產生『鑰匙格式』（KEYMAT），KEYMAT 是產生各種鑰匙的基本元件；至於如何由 KEYMAT 產生加密或認證金鑰，這可必須依照各種密碼系統而定。無論如何，雙方依照所交換鑰匙材料製造出 KEYMAT 之後，就可自行製造出其他鑰匙，並且可以填入 SA 記錄之中（SAD 資料庫）。依照不同的安全強度需求，製造 KEYMAT 有下列三種方法：

- **不採用『完全性順向機密』的安全機制**：表示會議金鑰是由一般性資料產生，而沒有任何交換鑰匙材料（KE），則新的鑰匙格式如下：

  KEYMAT = prf(SKEYID_d, protocol | SPI | Ni_b | Nr_b)

  其中，SKEYID_d 是由第一階段協商出來的；protocol 與 SPI（Security Parameter Index）兩者都是由 ISAKMP 封包上的『提案承載』（Proposal Payload）取得，其中 protocol 表示雙方的通訊協定（IPSec AH 或 IPSec ESP）；SPI 的產生就較

為複雜，如果僅是 ISAKMP 封包，則 SPI 是由 Initiator Cookie 與 Responder Cookie 兩個欄位所組成；如果是 IPSec 封包，則 SPI 是由 IPSec 標頭上的 SPI 欄位取得。

◈ **採用『完全性順向機密』的安全機制**：表示是經由雙方交換會議材料，再計算出來新的會議金鑰。當交換鑰匙材料之後，會依照 Diffie-Hellman 演算法計算出鑰匙基本元素：$g(gm)^{xy}$，新鑰匙格式如下：

KEYMAT = prf(SKEYID_d, $g(gm)^{xy}$ | protocol | SPI | Ni_b | Nr_b)

◈ **重複連結技巧**：在某些安全性要求較高的情況，可使用重複連結技巧來製作鑰匙格式，其製作技巧如下：(比使用單一個 prf 計算來得安全)

KEYMAT = K1 | K2 | K3 | …

其中：

K1 = prf(SKEYID_d, [$g(gm)^{xy}$] | protocol | SPI |Ni_b | Nr_b)

K2 = prf(SKEYID_d, K1 | [$g(gm)^{xy}$] | protocol | SPI |Ni_b | Nr_b)

K3 = prf(SKEYID_d, K2 | [$g(gm)^{xy}$] | protocol | SPI |Ni_b | Nr_b)

… 等等

如果採用短暫使用的 Diffie-Hellman (Ephemeral Diffie-Hellman) 機制，來交換鑰匙材料 ($g(gm)^{xy}$)，也就是說，所交換出來的鑰匙格式僅使用一次 (或較短暫的時間)；下一次通訊時，必須再重新協議新的鑰匙格式。在這種情況下，第一階段協商所產生的 SKEYID_e 與 SKEYID_a 將不會被使用於快速模式的協商之中，兩者僅提供於第一階

段的認證（SKEYID_a）與加密（SKEYID_e）使用。

在快速模式的運作程序中，也允許同時建立多個 SA 之間的鑰匙交換。如圖 10-25 所示，發起者傳送兩個 SA（SA0 與 SA1），而回應者也發送兩個 SA（SA0 與 SA1）給發起者，就每一個 SA 都是獨立且單方性的特性而言，其實它們之間已經建立了四個安全關聯。

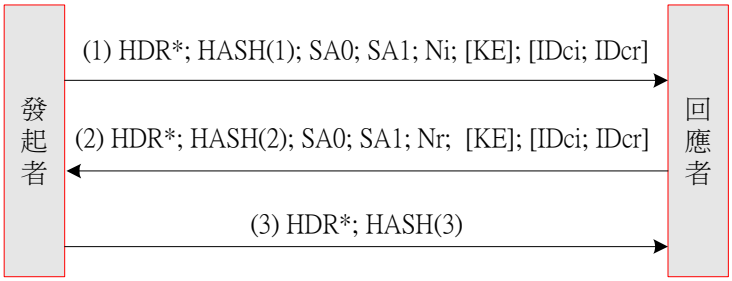


圖 10-25 快速模式的多重 SA 協商

(B) 新群組模式

『新群組模式』（New Group Mode）的運作時機是，當建立完成第一階段協商之後，雙方才可以協商建立新的群組，但這個時機是在還未進入第二階段協商之前，因此，它可以屬於第一階段或第二階段協商。但每一次執行新群組模式運作時，必須再重複執行第二階段交換鑰匙的程序，這是因為製作鑰匙材料已經改變，而必須經過交換程序再產生新的鑰匙。

就我們所知道，IKE 協定是利用 Diffie-Hellman 演算法來計算雙方所需的共享金鑰，而該演算法需要一個公開的亂數來作為計算鑰匙的基本元素（Prime）；為了符合協議此亂數的需求，Oakley 協定便定義了四個群組的基本元素，所以新群組模式主要是用來協議應該採用哪一群組，其運作程序如圖 10-26 所示。

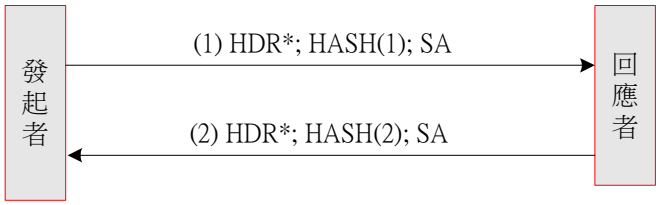


圖 10-26 新群組模式的運作程序

雙方協議群組模式的訊息是存放於 SA 的『提案承載』內，其中 HASH(1) 與 HASH(2) 的計算方法相同，如下：

HASH(1) = prf(SKEYID_a, M-ID | SA)

HASH(2) = prf(SKEYID_a, M-ID | SA)

上述中， HASH(1) 與 HASH(2) 是 prf 計算的輸出值，使用 ISAKMP 封包標頭上的訊息欄位（M-ID）與 SA 提案承載（包含標頭與承載）的連結值，及 SKEYID_a 經過某一雜湊演算法所計算得來的。

(C) ISAKMP 訊息交換

當 IPSec SA(快速模式)建立完成之後，便可以利用『ISAKMP 訊息交換』(ISAKMP Information Exchange) 協定來交換訊息；此交換模式與 ISAKMP SA 建立之後再交換訊息沒有兩樣，都是利用連線類別 5，只不過兩者交換訊息有所不同而已。其運作程序如圖 10-27 所示。

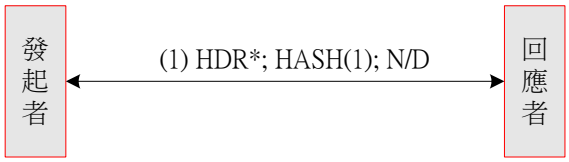


圖 10-27 ISAKMP 訊息交換的運作程序

圖 10-27 中，N/D 表示傳送 ISAKMP 通知承載(Notify Payload)或 ISAKMP 刪除承載 (Delete Payload)；HASH(1) 是 prf 計算的輸出值，其計算方法如下：

HASH(1) = prf(SKEYID_a, M-ID | N/D)

這裡必須強調一下，經過第二階段協商後，不同 ISAKMP SA 之間的訊息標示(M-ID)是不會重複的，而整個封包的承載是經由 SKEYID_e 所加密。

10-7 VPN 網路規劃與管理

10-7-1 VPN 網路規劃

(請匯入：VPN 網路_空白.pkt)

目前公司有兩個重要據點:高雄總公司與紐約分公司,兩公司內大約都有 100 部工作站。公司期望將兩地的網路透過網際網路結合一個虛擬私有網路，期望網路架構圖，如圖 10-28 所示。

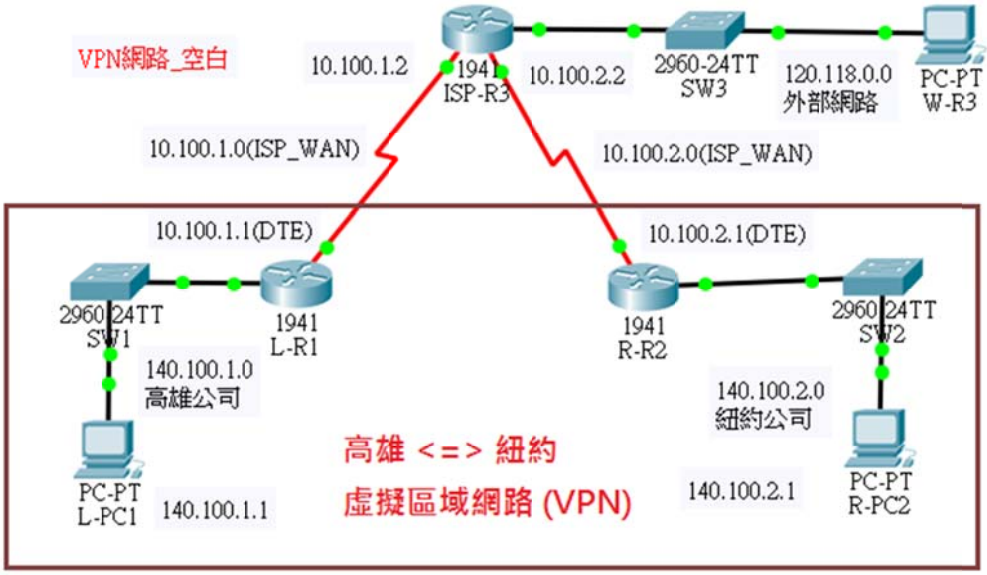


圖 10-28 VPN 網路範例規劃

(A) 網路概況

VPN 網路架構可區分為：本地網路、遠端網路與 ISP 網路等三大部分，VPN 兩網路透過廣域網路 (WAN) 的 Serial 連線銜接到 ISP 網路，兩地封包是透過 ISP 轉送到達目的地。網路架構說明如下：

- **高雄私有網路(140.100.1.0/24)**：由 Cisco 1941(L-R1) 路由器建置而成，由 L-PC1 主機(192.168.0.1) 代表本地主機。
- **紐約私有網路(140.100.2.0/24)**：由 Cisco 1941 路由器建置而成，由 R-PC2 (192.168.100.1) 主機代表遠端主機。
- **ISP 公眾網路**：由 Cisco 1941 路由器代表，並連接外部網路 (120.118.0.0/24)，並以 W-R3 (120.118.0.1) 主機代表。

(B) 網路規劃與建置

- 網路環境規劃

網路區段	Gateway/DNS	名稱	IP 位址	連結介面
140.100.1.0 255.255.255.0	140.100.1.254 168.95.1.1	L-PC1	140.100.1.1/24	SW1(Fa0/1)
140.100.2.0 255.255.255.0	140.100.2.254 168.95.1.1	R-PC2	140.100.2.1/24	SW2(Fa0/1)
120.118.0.0 255.255.255.0	120.118.0.1 168.95.1.1	W-PC3	120.118.0.1/24	SW3(Fa0/1)
10.100.1.0 255.255.255.0		ISP-WAN		
10.100.2.0 255.255.255.0		ISP-WAN		

- 路由器規劃：

路由器	網路區段	IP 位址	埠口	對應埠口
L-R1	140.100.1.0/24	140.100.1.254	Gi0/0	Gi0/1(SW1)
	10.100.1.0/24	10.100.1.1(DTE)	Se0/1/0	Se0/1/0(ISP-R3)
R-R2	140.100.2.0/24	140.100.2.254	Gi0/0	Gi0/0(SW2)
	10.100.2.0/24	10.100.2.1(DTE)	Se0/1/0	Se0/1/1(ISP-R3)

ISP-R3	10.100.1.0/24	10.100.1.2(DCE)	Se0/1/0	Se0/1/0(L-R1)
	10.100.2.0/24	10.100.2.2(DCE)	Se0/1/1	Se0/1/0(R-R2)
	120.118.0.0/24	120.118.0.254	Gi0/0	Gi0/1(SE3)

■ ISP-WAN 網路連線規劃：

網路	型態	介面	IP 位址	bandwidth	Clock rate
10.100.1.0/24	DCE	ISP-R3(se0/1/0)	10.100.1.2	10(1G)	56000
	DTE	L-R1(se0/1/0)	10.100.1.1	10 (1G)	
10.100.2.0/24	DCE	ISP-R3(Se0/1/1)	10.100.2.2	10 (1G)	56000
	DTE	R-R2(Se0/1/0)	10.100.2.1	10 (1G)	

■ 靜態繞路規劃：

依照圖 10-13，盡量讓網路內各主機可繞路成功，規劃如下：

Router	Destination AD	Network Mask	Net Hop	備註
L_R1	0.0.0.0	0.0.0.0	10.100.1.2	往外傳送
R_R2	0.0.0.0	0.0.0.0	10.100.2.2	往外傳送
ISP_R3	140.100.1.0	255.255.255.0	10.100.1.1	往內傳送
	140.100.2.0	255.255.255.0	10.100.2.1	往內傳送
	0.0.0.0	0.0.0.0	120.118.0.1	往外傳送

(C) L-R1 路由器網路設定

```
L-R1#config ter
L-R1(config)#int gi0/0
L-R1(config-if)#ip address 140.100.1.254 255.255.255.0
L-R1(config-if)#no shutdown
L-R1(config-if)#int s0/1/0
L-R1(config-if)#ip address 10.100.1.1 255.255.255.0
L-R1(config-if)#bandwidth 10
```

```
L-R1(config-if)#no shutdown
L-R1(config-if)#exit
L-R1(config)#ip route 0.0.0.0 0.0.0.0 10.100.1.2
L-R1(config)#do show ip int brief
      GigabitEthernet0/1 unassigned YES unset administratively down down
      Serial0/1/0 10.100.1.1 YES manual up up
      Serial0/1/1 unassigned YES unset administratively down down
      Vlan1 unassigned YES unset administratively down down
L-R1(config)#do show ip route
      Gateway of last resort is 10.100.1.2 to network 0.0.0.0
```

(D) R-R2 路由器網路設定

```
R-R2>en
R-R2#config ter
R-R2(config)#int gi0/0
R-R2(config-if)#ip address 140.100.2.254 255.255.255.0
R-R2(config-if)#no shutdown
R-R2(config-if)#int s0/1/0
R-R2(config-if)#ip address 10.100.2.1 255.255.255.0
R-R2(config-if)#bandwidth 10
R-R2(config-if)#no shutdown
R-R2(config-if)#exit
R-R2(config)#ip route 0.0.0.0 0.0.0.0 10.100.2.2
R-R2(config)#
```

(E) ISP-R3 路由器網路設定

```
ISP-R3#config ter
ISP-R3(config)#int gi0/0
ISP-R3(config-if)#ip address 120.118.0.254 255.255.255.0
ISP-R3(config-if)#no shutdown
ISP-R3(config-if)#int s0/1/0
ISP-R3(config-if)#ip address 10.100.1.2 255.255.255.0
ISP-R3(config-if)#bandwidth 10
ISP-R3(config-if)#clock rate 500000
ISP-R3(config-if)#no shutdown
ISP-R3(config-if)#int s0/1/1
ISP-R3(config-if)#ip address 10.100.2.2 255.255.255.0
ISP-R3(config-if)#bandwidth 10
ISP-R3(config-if)#clock rate 500000
ISP-R3(config-if)#no shutdown
ISP-R3(config-if)#exit
ISP-R3(config)#ip route 140.100.1.0 255.255.255.0 10.100.1.1
```

```
ISP-R3(config)#ip route 140.100.2.0 255.255.255.0 10.100.2.1
ISP-R3(config)#ip route 0.0.0.0 0.0.0.0 120.118.0.1
ISP-R3(config)#do show ip route

      Gateway of last resort is 120.118.0.1 to network 0.0.0.0

      10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
      C 10.100.1.0/24 is directly connected, Serial0/1/0
      L 10.100.1.2/32 is directly connected, Serial0/1/0
      C 10.100.2.0/24 is directly connected, Serial0/1/1
      L 10.100.2.2/32 is directly connected, Serial0/1/1
      120.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
      C 120.118.0.0/24 is directly connected, GigabitEthernet0/0
      L 120.118.0.254/32 is directly connected, GigabitEthernet0/0
      140.100.0.0/24 is subnetted, 2 subnets
      S 140.100.1.0/24 [1/0] via 10.100.1.1
      S 140.100.2.0/24 [1/0] via 10.100.2.1
      S* 0.0.0.0/0 [1/0] via 120.118.0.1

ISP-R3(config)#
```

(F) 網路繞路測試

(請匯入：VPN 網路_網路設定.pkt)

- L-PC1 ping 140.100.2.1
- L-PC1 ping 120.118.0.1
- R-PC2 ping 140.100.1.1
- W-R3 ping 140.100.2.1

10-7-2 ISAKMP 安全套件規劃

(A) ISAKMP Phase 1：協商參數

吾人依照 ISAKMP 與 IKE 協定，規劃安全套件如下表：

參數		L_R1	R_R2
鑰匙分配方法	Manual or **ISAKMP**	ISAKMP	ISAKMP

加密演算法	DES、3DES、**ASE**	AES	AES
雜湊演算法	MD5、**SHA-1**	SHA-1	SHA-1
認證方法	**Pre-shared keys**、RSA	Pre-share	Pre-share
鑰匙交換	DH Group1、**2**、5	DH 2	DH 2
IKE Lifetime	**86400** Sec. or less	86400	86400
ISAKMP Key	**(自行定義)**	**csuMIS**	**csuMIS**

(B) ISAKMP Phase 2：協商參數

吾人依照 ISAKMP 與 IKE 協定，規劃安全套件如下表：

參數	L_R1	R_R2
安全套件名稱 **(自行定義)**	**VPN-SET**	**VPN-SET**
安全套件內容：IPSec 協定	ESP-AES、 ESP-SHA-HMAC	ESP-AES、 ESP-SHA-HMAC
對方主機	R_R2	L_R1
對方 IP 位址	10.100.1.1	10.100.2.1
加密網路範圍	140.100.1.0/24	140.100.2.0/24
網路範圍名稱 **(自行定義)**	**VPN-MAP**	**VPN-MAP**
SA 建立方法	ipsec-isakmp	ipsec-isakmp

10-7-3 IPSec VPN 設定

接下來，我們設定 L-R1 與 R-R2 之間**『位置對位置』(Sit-to-Sit)** 之間的 IPSec VPN，使兩邊網路結合成一個區域網路，並保證之間傳送是經過安全保護著。但它們之間傳遞的封包也許會經過多個路由器轉送，本範例僅用 ISP-R3 取代，表示是經過 ISP 公眾網路並在沒有安全保護底下轉送。設定步驟如下：

步驟 1：啟動 L_R1 安全套件與規劃 IPSec 參數、

步驟 2：啟動 R_R2 安全套件與規劃 IPSec 參數、

步驟 3：驗證 IPSec VPN 功能。

(A) L_R1 的 IPSec VPN 設定

啟動安全套件

『安全技術套件』(Security Technology Package, STP) 授權需啟動，才能使用相關套件，操作如下：

```
L-R1>en
L-R1#show version
      …… [套件授權模組 C1900，並還未啟動]
          ----------------------------------------------
          Device#   PID                  SN
          ----------------------------------------------
          *0        CISCO1941/K9         FTX15245FF0

          Technology Package License Information for Module:'c1900'

          -----------------------------------------------------------------
          Technology    Technology-package          Technology-package
                        Current       Type          Next reboot
          -----------------------------------------------------------------
          ipbase        ipbasek9      Permanent     ipbasek9
          security      disable       None          None
          data          disable       None          None

          Configuration register is 0x2102

             [起啟動命令如下]
L-R1#config ter
L-R1(config)#license boot module c1900 technology-package securityk9
L-R1(config)#end
L-R1#copy running-config startup-config
L-R1#reload
      ….
L-R1#show version
      …….[顯示已啟動成功]
          ----------------------------------------------------------------
          Technology    Technology-package          Technology-package
                        Current       Type          Next reboot
          ----------------------------------------------------------------
          ipbase        ipbasek9      Permanent     ipbasek9
          security      securityk9    Evaluation    securityk9
          data          disable       None          None

          Configuration register is 0x2102
```

設定訊務範圍 - ACL

L-R1 的管制流量(Ineresting Traffic) 即是他所銜接的網路 140.100.1.0/24 網路區段流向 R-R2 所管轄網路 140.100.2.0/24 之間的訊息。如果 R-R2 的管制流量即是相反方向。我們用 Access List 將它設定完成，如下：

```
L-R1(config)#ip access-list extended VPN-Traffic
L-R1(config)# permit ip 140.100.1.0 0.0.0.255 140.100.2.0 0.0.0.255
```

設定 IKE Phase 1 參數

設定 Phase 1 相關參數如下：

```
L-R1(config)#crypto isakmp policy 10          [安全編號 10]
L-R1(config-isakmp)#encryption aes 256
L-R1(config-isakmp)#authentication pre-share
L-R1(config-isakmp)#group 2                   [DH Group 2 參數]
L-R1(config-isakmp)#lifetime 86400
L-R1(config-isakmp)#exit
L-R1(config)#crypto isakmp key csuMIS address 10.100.2.1
L-R1(config)#
```

設定 IKE Phase 2 參數

首先設定安全套件名稱為『VPN-SET』，並選擇採用 esp-aes (ESP Cipher) 與 esp-sha-hmac (ESP 訊息認證演算法)，並將設定的加入 Crypto Map (VPN-MAP) 密件套件內，再將它嵌入網路介面內 (Se0/1/0) ，如下：

```
L-R1(config)#crypto ipsec transform VPN-SET esp-aes esp-sha-hmac
                          [安全套件名稱 VPNSET，以及 ESP-AES 密碼系統]
L-R1(config)#crypto map VPN-MAP 10 ipsec-isakmp
                [建立範圍名稱 VPN-MAP，採安全標號 10 與 isakmp 方式溝通]
L-R1(config-crypto-map)#set peer 10.100.2.1
L-R1(config-crypto-map)#set transform-set VPN-SET
L-R1(config-crypto-map)#match address VPN-Traffic
L-R1(config-crypto-map)#exit
L-R1(config)#int se0/1/0
L-R1(config-if)#crypto map VPN-MAP
      *Jan 3 07:16:26.785: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON
L-R1(config-if)#
```

(B) R_R2 的 IPSec VPN 設定 (大致上與 L-R1 相同)

啟動安全套件

『安全技術套件』(Security Technology Package, STP) 授權需啟動，才能使用相關套件，操作如下：

```
R-R2>en
R-R2#show version
      …… [套件授權模組 C1900，並還未啟動]
          ----------------------------------------------
          Device#   PID                   SN
          ----------------------------------------------
          *0        CISCO1941/K9          FTX15245FF0

          Technology Package License Information for Module:'c1900'

          -----------------------------------------------------------------
          Technology    Technology-package          Technology-package
                        Current       Type          Next reboot
          -----------------------------------------------------------------
          ipbase        ipbasek9      Permanent     ipbasek9
          security      disable       None          None
          data          disable       None          None

          Configuration register is 0x2102

              [如果還未啟動，起啟動命令如下]
R-R2(config)#license boot module c1900 technology-package securityk9
R-R2(config)#exit
R-R2#copy running-config startup-config
R-R2#reload
….
R-R2#show version
      …….[顯示已啟動成功]
          -----------------------------------------------------------------
          Technology    Technology-package          Technology-package
                        Current       Type          Next reboot
          -----------------------------------------------------------------
          ipbase        ipbasek9      Permanent     ipbasek9
          security      securityk9    Evaluation    securityk9
          data          disable       None          None

          Configuration register is 0x2102
```

設定管制流量 - ACL

L-R1 的管制流量(Ineresting Traffic) 即是他所銜接的網路 140.100.1.0/24 網路區段流向 R-R2 所管轄網路 140.100.2.0/24 之間的訊息。如果 R-R2 的管制流量即是相反方向。我們用 Access List 將它設定完成，如下：

```
R-R2(config)#ip access-list extended VPN-Traffic
R-R2(config-ext-nacl)#permit ip 140.100.2.0 0.0.0.255 140.100.1.0 0.0.0.255
R-R2(config-ext-nacl)#exit
```

設定 IKE Phase 1 參數

設定 Phase 1 相關參數如下：

```
R-R2(config)#crypto isakmp policy 10
R-R2(config-isakmp)#encryption aes 256
R-R2(config-isakmp)#authentication pre-share
R-R2(config-isakmp)#group 2
R-R2(config-isakmp)#lifetime 86400
R-R2(config-isakmp)#exit
R-R2(config)#crypto isakmp key csuMIS address 10.100.1.1
```

設定 IKE Phase 2 參數

首先設定安全套件名稱為『**VPN-SET**』，並選擇採用 esp-aes (ESP Cipher) 與 esp-sha-hmac (ESP 訊息認證演算法)，並將設定的加入 Crypto Map (VPN-MAP) 密件套件內，再將它嵌入網路介面內 (Se0/1/0) ，如下：

```
R-R2(config)#crypto isakmp key csuMIS address 10.100.1.1
R-R2(config)#crypto ipsec transform VPN-SET esp-aes esp-sha-hmac
R-R2(config)#crypto map VPN-MAP 10 ipsec-isakmp
      % NOTE: This new crypto map will remain disabled until a peer
      and a valid access list have been configured.
R-R2(config-crypto-map)#set peer 10.100.1.1
R-R2(config-crypto-map)#set transform-set VPN-SET
R-R2(config-crypto-map)#match address VPN-Traffic
R-R2(config-crypto-map)#exit
R-R2(config)#int s0/1/0
R-R2(config-if)#crypto map VPN-MAP
      *Jan 3 07:16:26.785: %CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON
```

(C) 驗證通訊連線

L-PC1 (140.100.1.1) 測試：

> ping 140.100.2.1 (R-PC2) **[OK]**

> ping 120.118.0.1 (W-R3) **[OK]**

R-PC2 (140.100.2.1) 測試：

> ping 140.100.1.1 (L-PC1) **[OK]**

> ping 120.118.0.1 (W-R3) **[OK]**

R-R3 (120.118.0.1) 測試：

> ping 140.100.2.1 (R-PC2) **[OK]**

> ping 140.100.1.1 (L-PC1) **[OK]**

10-7-4 驗證 IPSec VPN 傳輸

(A) 觀察產生 IPSec SA(安全關聯)

步驟 1：於 L-R1 上觀察 ipsec sa

```
L-R1#show crypto ipsec sa

    interface: Serial0/1/0
    Crypto map tag: VPN-MAP, local addr 10.100.1.1

    protected vrf: (none)
    local ident (addr/mask/prot/port): (140.100.1.0/255.255.255.0/0/0)
    remote ident (addr/mask/prot/port): (140.100.2.0/255.255.255.0/0/0)
    current_peer 10.100.2.1 port 500
    PERMIT, flags={origin_is_acl,}
    #pkts encaps: 0, #pkts encrypt: 0, #pkts digest: 0
    #pkts decaps: 0, #pkts decrypt: 0, #pkts verify: 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0
    …
```

步驟 2：於 L-PC1 ping R-PC2 讓 SA 產生動作

```
C:\>ping 140.100.2.1                    [OK]
```

步驟 3：再觀察 L-R1 的 ipsec sa

```
L-R1#show crypto ipsec sa

    interface: Serial0/1/0
    Crypto map tag: VPN-MAP, local addr 10.100.1.1
```

```
protected vrf: (none)
local ident (addr/mask/prot/port): (140.100.1.0/255.255.255.0/0/0)
remote ident (addr/mask/prot/port): (140.100.2.0/255.255.255.0/0/0)
current_peer 10.100.2.1 port 500
PERMIT, flags={origin_is_acl,}
#pkts encaps: 3, #pkts encrypt: 3, #pkts digest: 0        [已產生 IPSec 封包]
#pkts decaps: 2, #pkts decrypt: 2, #pkts verify: 0
#pkts compressed: 0, #pkts decompressed: 0
#pkts not compressed: 0, #pkts compr. failed: 0
#pkts not decompressed: 0, #pkts decompress failed: 0
#send errors 1, #recv errors 0

local crypto endpt.: 10.100.1.1, remote crypto endpt.:10.100.2.1
path mtu 1500, ip mtu 1500, ip mtu idb Serial0/1/0
current outbound spi: 0x11E25D0F(300047631)
```

(B) 觀察 IPSec SA 相關設定 (由 L-R1 操作)

查詢 IPSec Transform-set

```
L-R1#show crypto ipsec transform-set
    Transform set VPN-SET: { { esp-aes esp-sha-hmac } }
    will negotiate = { Tunnel, },

    Transform set #$!default_transform_set_1: { esp-aes esp-sha-hmac }
    will negotiate = { Transport, },
    Transform set #$!default_transform_set_0: { esp-3des esp-sha-hmac }
    will negotiate = { Transport, },
```

查詢 isakmp sa

```
L-R1#show crypto isakmp sa
    IPv4 Crypto ISAKMP SA
    dst             src             state          conn-id  slot      status
    10.100.2.1      10.100.1.1      QM_IDLE          1019      0      ACTIVE
```

查詢 isakmp policy

```
L-R1#show crypto isakmp policy
    Global IKE policy
    Protection suite of priority 10
        encryption algorithm: AES - Advanced Encryption Standard (256 bit keys).
        hash algorithm: Secure Hash Standard
```

```
        authentication method: Pre-Shared Key
        Diffie-Hellman group: #2 (1024 bit)
        lifetime: 86400 seconds, no volume limit
```

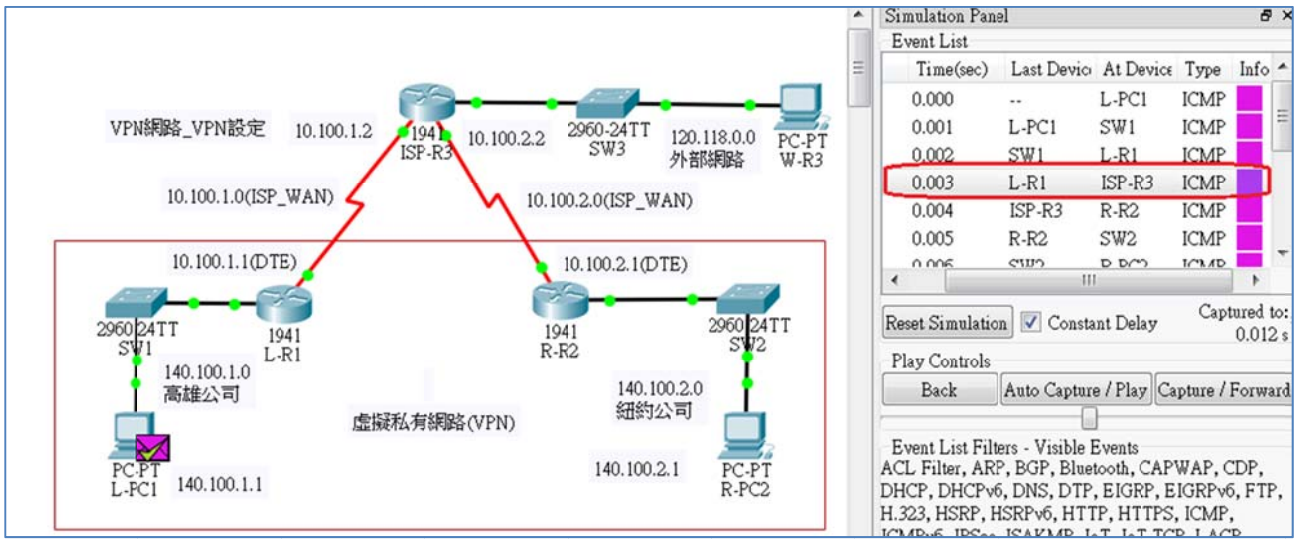
查詢 crypto map

```
L-R1#show crypto map
    Crypto Map VPN-MAP 10 ipsec-isakmp
        Peer = 10.100.2.1
        Extended IP access list VPN-Traffic
            access-list VPN-Traffic permit ip 140.100.1.0 0.0.0.255 140.100.2.0
            0.0.0.255
        Current peer: 10.100.2.1
        Security association lifetime: 4608000 kilobytes/3600 seconds
        PFS (Y/N): N
        Transform sets={
            VPN-SET,
        }
        Interfaces using crypto map VPN-MAP:
            Serial0/1/0
```

(C) 觀察 IPSec 封包內的 ESP 標頭

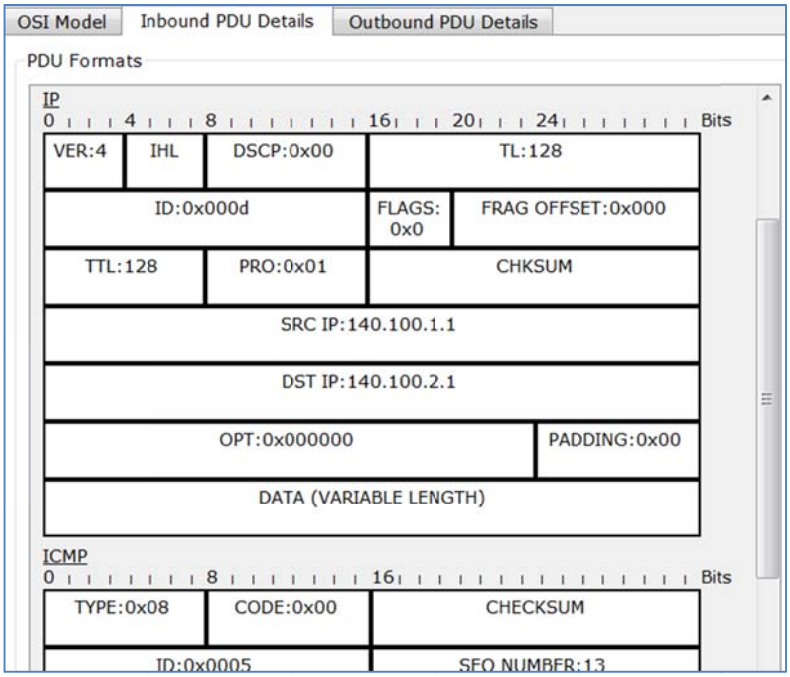
由 L-PC1 ping R-PC2：

將 Packet Tracer 設定成 Simulation Mode，再由 L-PC1 ping R-PC2，觀察封包進出 L-R1 路由器之前與之後之封包，如下：



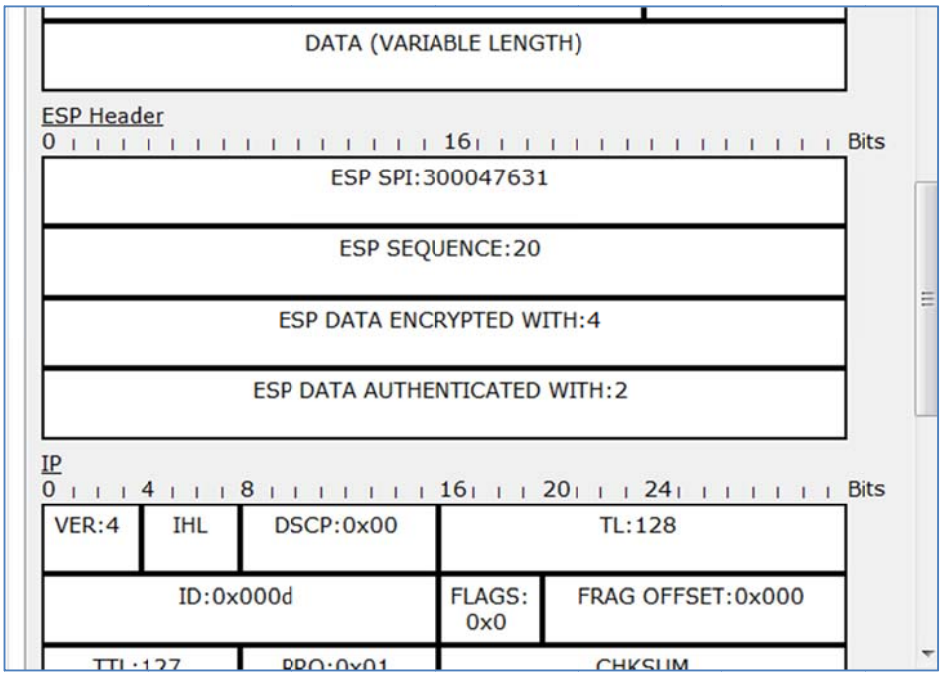
擷取進入 L-R1 之前的 IP 封包

封包沒有經過 IPSec ESP 包裝的原始 IP 封包。



擷取離開 L-R1 之後的 IP 封包

IP 封包經過 IPSec ESP 包裝後型態。



(D) 觀察 ISAKMP 封包協定運作

請自行練習

10-8 專題研討：Firewall-VPN 網路規劃

10-8-1 Firewall-VPN 網路規劃

(A) 系統需求

(請匯入 Firewall-VPN 網路_空白.pkg)

目前公司有兩個重要據點:高雄總公司與越南分公司,兩公司內大約都有 100 部工作站。公司期望將兩地的網路透過網際網路結合一個虛擬私有網路，如圖 10-29 所示。除了將兩網路整合成 VPN 網路外，每一區域網路也須保留與外部網路的通訊能力並具有防火牆功能，如下：

1. 禁止外部主機存取內部任何資源(進入內部網路)。
2. DNZ 網路(140.100.200.0/24 或 140.102.200.0/24) 上主機可以與內部網路主機相互連線。
3. 允許外部可以任意存取 DNZ 網路內伺服器(如 L-Outer-HTTP 或 R-Outer-HTTP)。
4. 允許外部網路可以測試 (Ping) DNZ 網路下主機。
5. 允許內部主機可以任意存取外部網路資源。

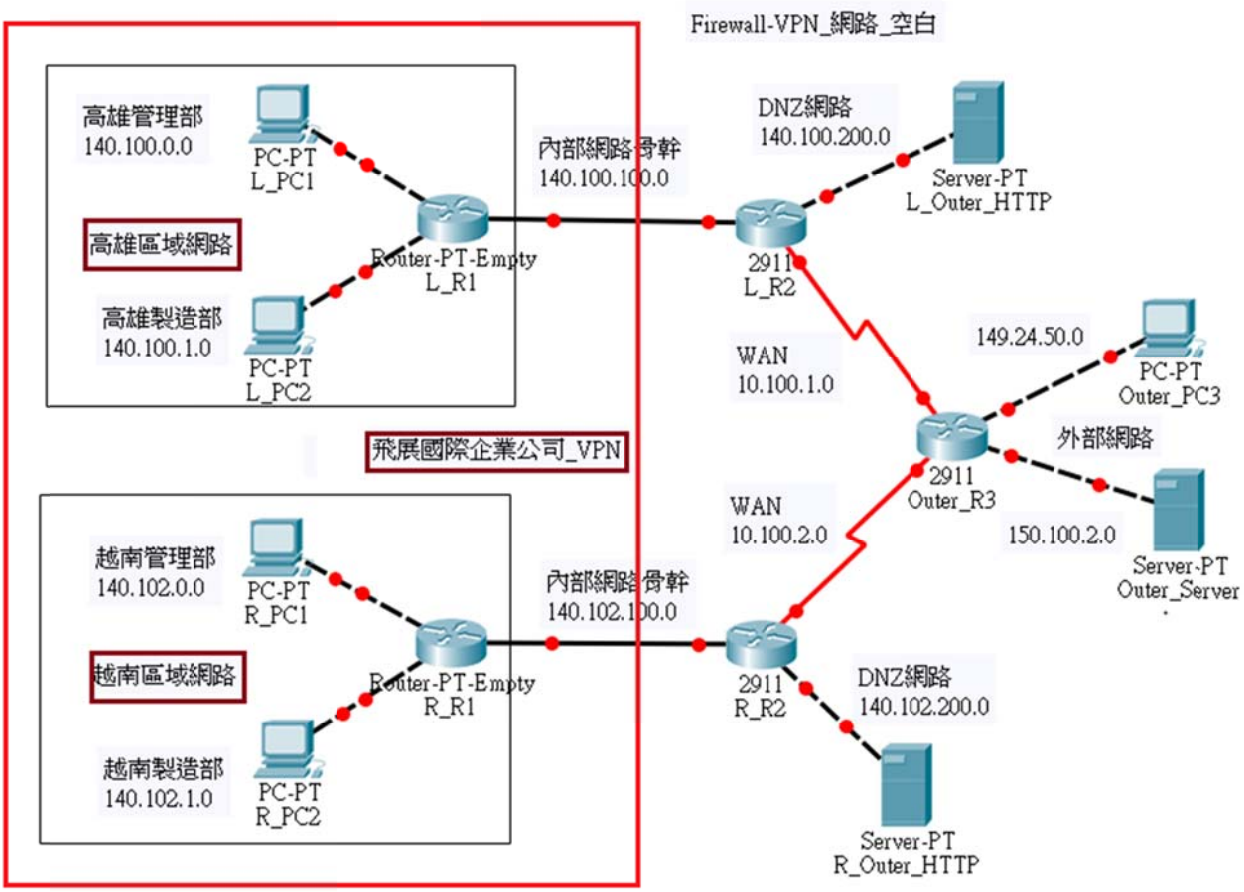


圖 10-29 Firewall-VPN 網路規劃

(B) 網路環境規劃

圖 10-29 網路環境的規劃如下：

地區	網路區段	gateway	路由介面	主機代表	IP 位址
高雄區域網路區域	140.100.0.0/24	140.100.0.254	L-R1(Gi0/0)	L-PC1	140.100.0.1
	140.100.1.0/24	140.100.1.254	L-R1(Gi1/0)	L-PC2	140.100.1.1
	140.100.200.0/24 DNZ 網路	140.100.200.254	L-R2(Gi2/0)	L-Outer-http	140.100.200.1
	140.100.100.0		L-R1(Gi3/0)		140.100.100.1
			L-R2(Gi0/0)		140.100.100.2
越南區域	140.102.0.0/24	140.102.0.254	R-R1(Gi0/0)	R-PC1	140.102.0.1

網路	140.102.1.0	140.102.1.254	R-R1(Gi1/0)	R-PC2	140.102.1.1
	140.102.200.0/24 DNZ 網路	140.102.200.254	R_R2(Gi2/0)	R-Outer-http	140.102.200.1
	140.102.100.0		R-R1(Gi3/0)		140.102.100.1
			R-R2(Gi0/0)		140.102.100.2
外部網路	149.24.50.0/24	149.24.50.254	O-R3(Gi0/0)	Outer-PC3	149.24.50.1
	150.100.2.0/24	150.100.2.0.254	O-R3(Gi0/1)	Outer-Ser.	150.100.2.1

(C) 路由器規劃

本系統各區域的路由器介面卡規劃如下：

區域	Router	Router port	IP 位址	PC
高雄區域	L_R1 內部路由器	Gi0/0	140.100.0.254	L-PC1
		Gi1/0	140.100.1.254	L-PC2
		Gi3/0	140.100.100.1	L-R2(Gi0/0)
	L-R2 外部路由器	Gi0/0	140.100.100.2	L-R1(Gi3/0)
		Gi0/1	140.100.200.254	L-Out-HTTP
		Se0/2/0	10.100.1.1	Outer-R3(se0/2/0)
越南區域	R-R1 內部路由器	Gi0/0	140.102.0.254	R-PC1
		Gi1/0	140.102.1.254	R_PC2
		Gi3/0	140.102.100.1	R-R2(Gi0/0)
	R-R2 外部路由器	Gi0/0	140.102.100.2	R-R1(Gi3/0)
		Gi0/1	140.102.200.254	R-Out-HTTP
		Se0/2/0	10.100.2.1	Outer-R3(se0/2/1)
外部網路	R-R3 ISP 路由器	Gi0/0	149.24.50.254	Outer-PC3
		Gi0/1	150.100.2.1	Outer-Server
		Se0/2/0	10.100.1.2(DCE)	L-R2(Se0/2/0)
		Se0/2/1	10.100.2.2(DCE)	R-R2(Se0/2/0)

(D) 靜態路由表規劃

本系統各區域的路由器介面卡規劃如下：

地區	Router	Destination AD	Network Mask	Net Hop
高雄區域網路	L_R1	140.100.200.0	255.255.255.0	140.100.100.2
		0.0.0.0	0.0.0.0	140.100.100.2
	L_R2	140.100.0.0	255.255.255.0	140.100.100.1
		140.100.1.0	255.255.255.0	140.100.100.1
		0.0.0.0	0.0.0.0	10.100.1.2
越南區域網路	R_R1	140.102.200.0	255.255.255.0	140.102.100.2
		0.0.0.0	0.0.0.0	140.102.100.2
	R-R2	140.102.0.0	255.255.255.0	140.100.100.1
		140.102.1.0	255.255.255.0	140.100.100.1
		0.0.0.0	0.0.0.0	10.100.2.2
外部網路	Outer-R3	140.100.0.0	255.255.0.0	10.100.1.1
		140.102.0.0	255.255.0.0	10.100.2.1

10-8-2 網路介面與路由表設定

(請自行練習)

10-8-3 內部路由器 DNZ 設定

(請設定 L-R1 與 R-R1)

10-8-4 外部路由器 DNZ 設定

(請設定 L-R2 與 R-R2，自行練習)

10-8-5 VPN 安全套件規劃

(請自行練習)

10-8-6 IPSec VPN 設定

(請設定 L-R2 與 R-R2，自行練習)