

# 整合位址轉換之 IPv6 DNS 代理解析系統實作與應用

## Integrated Solution for IPv6 DNS Proxy Resolution and Address Translation on Windows Platform

張林煌<sup>1</sup> 莊坪達<sup>1</sup> 廖明沂<sup>2</sup> 羅有隆<sup>3</sup>

朝陽科技大學網路與通訊研究所<sup>1</sup> 朝陽科技大學資訊工程系<sup>2</sup> 朝陽科技大學電算中心<sup>3</sup>  
lchang@cyut.edu.tw

### 摘要

隨著台灣學術性網路之逐漸蓬勃發展，原生 (native)IPv6 連線資源已漸漸的從實驗環境走入一般使用環境中。原生 IPv6 連線提供使用者可以直接使用且應用 IPv6 網路協定之特色，例如更多之網路位址以及 QoS 等。然而在更進一步推廣使用 IPv6 的同時，微軟現行之主要作業系統，如 Windows 2000 與 Windows XP 等雖支援 IPv6 協定，但卻無法以 IPv6 透過 DNS 進行網域名稱解析，如此將造成在使用者不便以及推廣上之困難。因此，在本論文中，我們將開發一 DNS 代理查詢程式以解決此一問題。此系統並搭配位址轉換程式，一併解決與整合純 IPv6 系統在連接 IPv4 網路資源時所造成之問題。

**關鍵詞：**IPv6、網域名稱、代理解析、位址轉換

### Abstract

The native IPv6 connection is getting widely deployed within academic network provider in Taiwan, such as TAnet2 and TWAREN. The native IPv6 connection provides end-users to realize the IPv6 features. However, some end-user operation systems, such as Microsoft Windows 2000 and Windows XP, do not support DNS query operation via IPv6 connections. It may cause inconvenient to the users and hinder IPv6 connection to end users from popularity. Thus, we propose and develop a system that provides DNS query proxy to solve these problems under Windows platforms.

**Keywords:** IPv6, DNS, Proxy Resolution, Network Address Translation

### 1. 前言

IPv6(Internet Protocol version 6)發展至今已近十年，近年來 IPv6 在網路建置上並沒有明顯的進展，起因於除了網路硬體的支援問題等管理面主題以外，使用者端之環境與應用程式，亦是造成 IPv6 遲遲無法大量普及原因之一。

目前已有許多 IPv6 與 IPv4 過度時期之轉換或共存機制，如 NA(P)T-PT (Network Address

Translation - Protocol Translation) [5]、DSTM (Dual Stack Transition Mechanism) [2]、6to4 [3]、ISATAP (Intra-Site Automatic Tunnel Addressing Protocol) [6] 與 Tunnel[1]等。由於應用場合與時機的不同，所使用之機制亦有所不同。

IPv6 近年來在使用者角度所探討之主題，除了使用 Tunnel 機制提供 IPv4 使用者可存取 IPv6 之網路資源，但此方式因需透過 tunnel broker 進行封包之轉送，因此在網路存取速度上較為緩慢。此外即為雙堆疊(Dual Stack)之雙協定共存實施方式，意即使用者之所使用系統同時具有 IPv4 與 IPv6 連線資源，反之使用者保留 IPv4 連線資源，相對亦消耗有限之 IPv4 資源。因此在適當之環境，例如上游連線單位可提供 Native IPv6 連線資源下，提供純 IPv6(pure IPv6)之連線資源，而成了較為恰當之作法。然而由於以往支援 IPv6 協定之應用程式與服務仍相當有限，使得比較少受到重視與討論。

近年來由於 IPv6 之逐漸發展，使得 IPv6 之路由以日趨完整。另外一方面，台灣高品質學術研究網路(Taiwan Advanced Research & Education Network, TWAREN)提供連線單位原生 IPv6 連線與完整之路由，因此在建置純 IPv6 環境上，提供了絕佳的研究環境

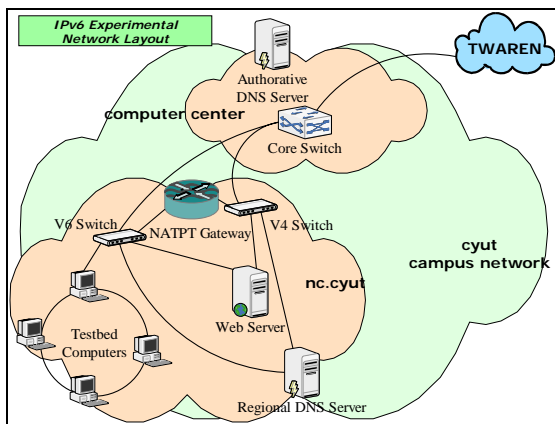
微軟現有之作業系統中，Windows 2000、Windows XP 與 Windows 2003 Server 均已內建或可透過更新之方式支援 IPv6 協定。然而以上之作業系統中，Windows 2000 與 Windows XP 尚無法支援以 IPv6 封包傳送 DNS 訊息，意即在使用純 IPv6 之主機上，無法透過 DNS 解析網域名稱，僅能以 IPv6 位址進行連線，造成使用者使用上之不便與推廣上之問題。

因此，在本論文中，我們將設計提供應用程式進行 DNS 查詢之代理系統，以透過本系統向指定或透過自動查詢網路上之 DNS Server 進行網域名稱解析。另外，由於目前網路資源絕大部分仍採用 IPv4，因此若查詢得知 IPv4 之位址，對於純 IPv6 之系統，由於並無法直接連線，因此需透過網路協定轉換(NAT-PT)，用以連線至 IPv4 網路資源。因此本系統亦包含一 DNS 轉換程式，提供將 IPv4 位址

透過 RFC 3056 中之定義，轉換為一暫時之 6to4 IPv6 位址，搭配網路 NAT-PT 閘道器進行協定轉換與封包轉送。

## 2. 網路架構

在網路架構方面，TWAREN 於 93 學年度起支援 IPv6 協定，並提供連線學校使用 IPv6 連線，以推廣 IPv6 之使用。而本校亦於 94 年 2 月連接至 TWAREN，而為著眼於未來 IPv6 之推廣與實驗網路環境平台建置，本校並於同年由網路與通訊研究所與電子計算機中心合作下一世代網路環境建置計畫－IPv6 校園化。藉以透過本計畫之執行，使得本校網通所成為 IPv6 之試行連線單位，提供相關之服務以及使用者推廣等研究。



圖一 校園 IPv6 實驗網路架構圖

本計畫之連線架構圖如一所示。在校外連線部分，本校電算中心透過 Cisco 6509 與中部區網中心連接 TWAREN。校內部份，IPv6 試行單位為 IPv4 與 IPv6 獨立路由規劃，IPv6 連線透過連線單位之交換器直接連至電算中心之 Core Switch，如此可保留原有之 IPv4 網路架構提供較穩定之並存環境。

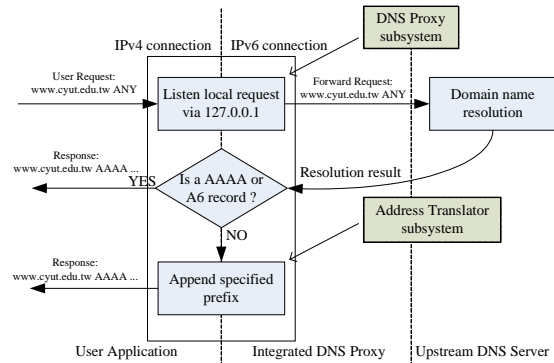
服務方面，本校校內連線單位網通所將現有之網路服務，如 WWW、FTP 與 DNS 等，均加上 IPv6 支援。DNS 部分亦完成校內分配 IPv6 網段之正反解查詢。下一階段之服務將以 SIPv6、MPLS over IPv6 等服務為建置目標。

測試實驗環境方面，本論文中所開發建置之系統測試環境(testbed)為純 IPv6 連線主機，直接與 IPv6 Switch 連線。而在同一 IPv6 網段上亦建置網路協定轉換(Network Address Translation - Protocol Translation, NAT-PT)閘道器，用以提供 IPv6 使用者存取 IPv4 資源。

## 3. 系統設計

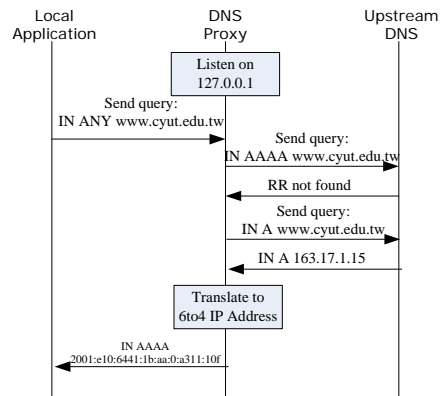
如上所述，本系統包含兩部分：一、DNS 代理查詢系統(DNS Proxy Subsystem)，二、網路位址

轉換系統(Address Translation Subsystem)。系統架構如圖二所示，DNS 代理查詢系統透過 IPv4 介面，接受來自本機端(localhost)之 DNS 請求，收到來自應用程式之網域名稱解析請求之後，將透過 IPv6 連線，將請求轉送至指定之上游(upstream)DNS 伺服器，進行名稱解析，流程示意圖請參照圖三。



圖二 系統架構圖

經過解析之後之結果將回傳至代理查詢系統，由代理查詢系統判斷是否再次進行遞迴解析(recursively resolution)。若查詢結果中具有 IPv6 位址(AAAA 紀錄)，則將該結果回傳給予使用者應用程式。但若查詢結果中僅回傳 IPv4 位址(A 紀錄)，則將該位址傳送至網路位址轉換系統，將 IPv4 位址轉換成為唯一之 6to4 IPv6 位址，並回傳予應用程式，其演算虛擬碼如圖四所示。



圖三 流程示意圖

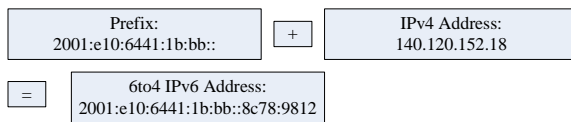
在網路位址轉換子系統部分，由於在純 IPv6 系統上，並沒有 IPv4 連線與路由，因此 IPv4 位址並不具有實際意義。因此若 DNS 查詢得到遠端主機之 IPv4 位址，將必須要透過位址轉換成 IPv6 位址，並搭配 NAT-PT 伺服器進行網路協定轉換(protocol translation)。其運作原理為，NAT-PT 伺服器將設定一前置位址(address prefix)，此前置位址必須唯一且不與現行網路位址相衝突。而查詢所得之 IPv4 位址將分別轉換成十六進位之表示方式，且附加於前置位址之後。

```

If ( response == CNAME )
  Send Recursive resolution to upstream DNS
Else if ( response == AAAA )
  Send response to application
Else if ( response == A )
  Translate to AAAA
  Send response to application
....

```

以主機 tanet2005.nchu.edu.tw 為例，其 IPv4 位址為 140.120.152.18，將其 140 與 120 轉換成為十六進位而成為 8c 與 78，後兩位 152 與 18 則同樣轉換而成為十六進位之 98 與 12，並將得到之四組十六進位字元，置於 NAT-PT prefix 之後，其中不足長度之位元則補零，運作機制如圖五。



圖五 6to4 位址轉換示意圖

NAT-PT 伺服器在接受到此一特殊位址之後，將會透過相同之方法，轉換回原有之 IPv4 位址並進行連線，如此內部之純 IPv6 系統主機即可透過此機制與 IPv4 網路資源進行存取。

#### 4. 執行結果

本節將針對系統實際應用結果來說明，首先圖六為本系統之執行畫面，此介面可提供使用者設定本 DNS Proxy 之相關參數，如 NAT-PT prefix、本地端之 DNS 埠號，以及轉送之上游 DNS Server。使用者除了可自行設定上游 DNS Server 外，並可透過 IPv6 協定中之 neighbor solicitation 機制自動查詢位於該網段之 DNS 伺服器。應用程式介面亦可得知目前伺服器執行狀況與程式紀錄。

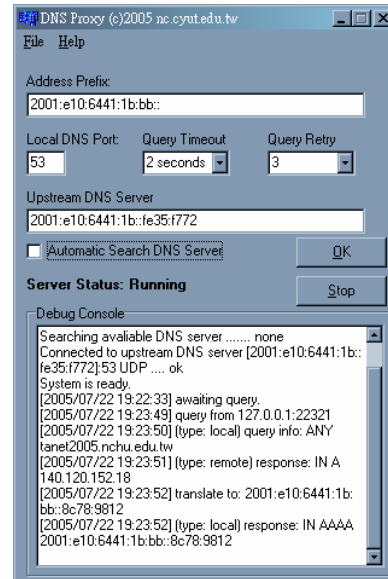
在應用程式執行方面，我們將透過數個系統內建指令(nslookup, ping, tracer)針對一僅提供 IPv4 連線之主機，進行連線，以測試本系統之功能。在此範例中，我們分別採用 www.kame.net 作為 IPv6 連線測試主機，以及採用 tanet2005.nc hu. edu. tw 作為 IPv4 連線與 DNS 轉換測試主機。

圖七為 DNS 解析測試結果，由圖中我們可以得知，當系統進行 DNS 解析後，得到查詢網域名稱擁有 IPv6 位址時，則將其 IPv6 位址回傳給與應用程式。反之若網域名稱僅有 IPv4 位址，則將啟動轉換機制，將其 IPv4 位址轉換而成 6to4 IPv6 位址，並回傳予應用程式。

另外，圖八為連線測試結果。我們透過 ping

指令針對測試主機 tanet2005.nchu.edu.tw 進行測試，結果 ping 程式正確的解析出其 6to4 IPv6 位址，並正確的進行 ICMPv6 echo 請求。

最後則是網路路由追蹤結果，如圖九所示。我們針對測試主機進行路由追蹤，以觀察其連線經過之路由。由結果中我們可觀察到，其路由經過 NAT-PT 開道器之後(圖中 2001:e10:6441:1b:aa::1)，接下來之路由均透過 IPv4 路由進行。然而其位址均由 NAT-PT 開道器轉換成為 6to4 IPv6 位址。



圖六 系統介面

```

C:\>nslookup
Default Server: localhost
Address: 127.0.0.1

> set type=aaaa
> www.kame.net
Server: localhost
Address: 127.0.0.1

Non-authoritative answer:
www.kame.net AAAA IPv6 address = 2001:200:0:8002:203:47ff:fea5:3085

> tanet2005.nchu.edu.tw
Server: localhost
Address: 127.0.0.1

Non-authoritative answer:
tanet2005.nchu.edu.tw AAAA IPv6 address = 2001:e10:6441:1b:bb:0:8c78:9812

```

圖七 nslookup (DNS 解析) 結果

```

C:\>ping tanet2005.nchu.edu.tw

Pinging tanet2005.nchu.edu.tw [2001:e10:6441:1b:bb:0:8c78:9812] with 32 bytes of data:

Reply from 2001:e10:6441:1b:bb:0:8c78:9812: time=4ms
Reply from 2001:e10:6441:1b:bb:0:8c78:9812: time=3ms
Reply from 2001:e10:6441:1b:bb:0:8c78:9812: time=2ms
Reply from 2001:e10:6441:1b:bb:0:8c78:9812: time=2ms

Ping statistics for 2001:e10:6441:1b:bb:0:8c78:9812:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 4ms, Average = 2ms

```

圖八 ping (連線測試) 結果

經由以上之測試，我們透過此一系統，結合區域網路上之 NAT-PT 開道器，即可提供使用視窗平台之原生 IPv6 使用者存取 IPv4 支援之整合式解決方案。

```

C:\>tracert tanet2005.nchu.edu.tw

Tracing route to tanet2005.nchu.edu.tw [2001:e10:6441:1b:bb:0:8c78:9812]
over a maximum of 30 hops:

  0  <1 ms    <1 ms    <1 ms    2001:e10:6441:1b:aa::1
  1  <1 ms    <1 ms    <1 ms    2001:e10:6441:1b::1
  2  1 ms     1 ms     1 ms     2001:e10:6441:1b:bb:0:a311:cfcd
  3  2 ms     1 ms     2 ms     2001:e10:6441:1b:bb:0:a311:7fa
  4  2 ms     3 ms     3 ms     2001:e10:6441:1b:bb:0:a311:7bd
  5  5 ms     3 ms     4 ms     2001:e10:6441:1b:bb:0:8c80:fb0e
  6  5 ms     2 ms     1 ms     2001:e10:6441:1b:bb:0:8c80:f8ca
  7  2 ms     4 ms     2 ms     2001:e10:6441:1b:bb:0:8c78:2f4d
  8  4 ms     2 ms     2 ms     2001:e10:6441:1b:bb:0:8c78:9812

Trace complete.
C:\>

```

圖九 traceroute (路由追蹤) 結果

## 5. 結論

推行 IPv6 使用時所遇到之問題，除了應用程式尚無普遍支援 IPv6 協定之外，IPv6 使用者無法存取 IPv4 之網路資源亦是需要面對之問題。然而在一般使用者大部分所使用之微軟視窗平台，無法支援使用者利用 IPv6 進行 DNS 查詢，則造成了在推行 IPv6 普及時相當不便與阻力。因此，透過以上所提出與開發之系統，透過本地端執行之程式，一併解決了連線以及 DNS 查詢之問題，使得使用者得以暢行無阻的使用網路資源。

在未來之研究方面，除了增進系統之穩定以外，在功能上我們將除了現有之正解對應轉換，亦將加上反解之對應，亦即透過本系統與原有之 IPv4 反解紀錄(Reverse PTR Lookup)與產生之 6to4 IPv6 位址進行對應，以健全本系統之功能。且本系統亦能在短期之內，以開放原始碼方式進行發表，期望能進一步的帶起 IPv6 一般使用者之環境建置與系統應用。

## 6. 致謝

本計劃之執行承蒙本校電子計算機中心及網路組李政儒先生與謝閔杰先生協助，特此致謝。

## 參考文獻

- [1] A. Durand, P. Fasano, I. Guardini, D. Lento, "IPv6 Tunnel Broker", RFC 3053, Jan. 2001.
- [2] J. Hagino, K. Yamamoto, "An IPv6-to-IPv4 Transport Relay Translator", IETF RFC 3142, June 2001.
- [3] B. Carpenter, K. Moore, "Connection of IPv6 Domains via IPv4 Clouds", IETF RFC3056, Feb. 2001.
- [4] P. Mockapetris, "Domain Names implementation and specification", IETF RFC 1035, Nov. 1987.
- [5] G. Tsirtsis, P. Srisuresh, "Network Address Translation - Protocol Translation", IETF RFC 2766, Feb. 2000
- [6] F. Templin, T. Gleeson, et. al., "Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)", IETF Draft draft-ietf-ngtrans-isatap-24.txt, Jan. 2005.
- [7] K. Tsuchiya, H. Higuchi, Y. Atarashi, "Dual Stack Hosts using the "Bump-In-the-Stack" Technique (BIS)", IETF RFC 2667, Feb. 2000.
- [8] KAME Project, <http://www.kame.net>
- [9] Vermicelli Project, "Trick or Treat Daemon", <http://www.vermicelli.pasta.cs.uit.no/ipv6/software.html>