Chapter 5.

Performance Analysis

5.1. Analysis on Various Methods of Address Resolution

We have a previous discussion on various methods of address resolution in section 4.5 that introduced three different address resolving policies for NAT-PT implemented on different platforms. However, different policies may result in obvious diversity in steps of connection establishment. Pre-resolution adopted as the policy of address resolution in Chassis-Based NAT-PT will bring the fastest way in establishing a connection between a pair of IPv4 and IPv6 node. We are giving an example of establishing a HTTP connection to prove that using the three different manners of address resolution with different platform right away. The concept of these three methods is going to be abridged since it has been detail described in previous chapter.

- Proxy Resolution in PC-Based NAT-PT (ordinary Linux PC)

The concept of Proxy Resolution is derived from Proxy ARP in IPv4 network. Figure 5.1 demonstrates an example of browsing an IPv6 Web server by an IPv4 host.

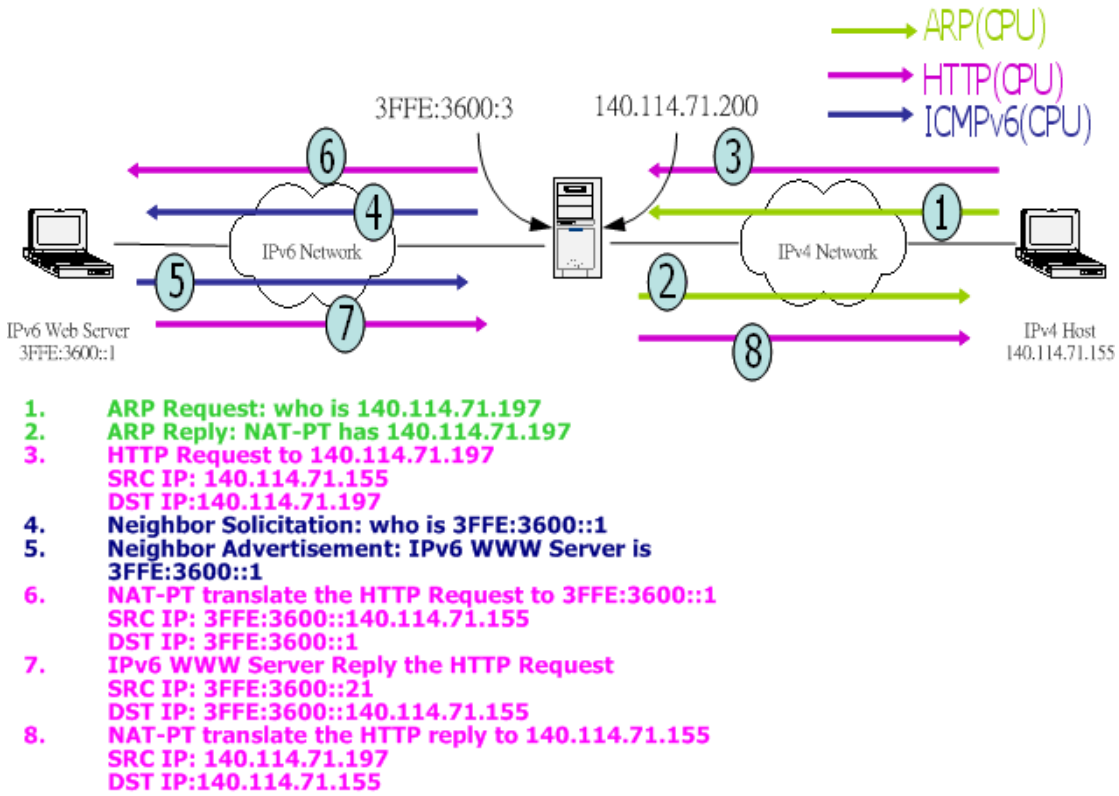
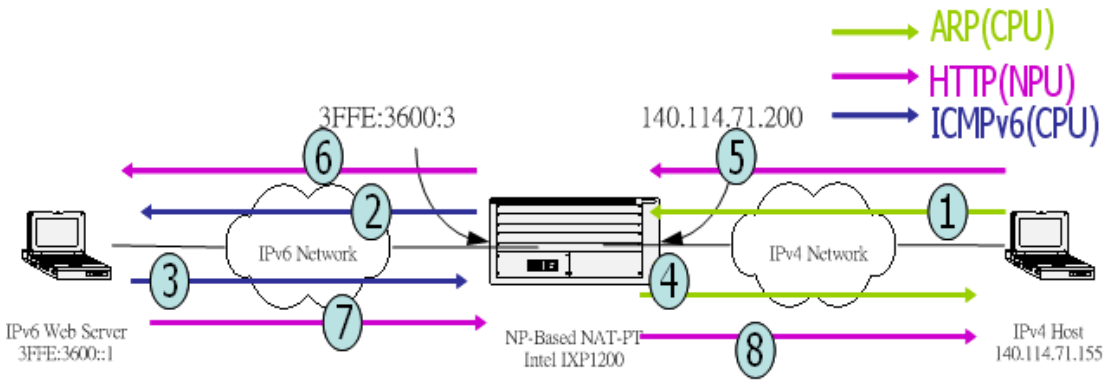


Figure 5.1 Example of Proxy Resolution in PC-Based NAT-PT

- Transparent Resolution in NP-Based NAT-PT (Intel IXDP1200)

Transparent Resolution takes the layer 2 address resolution as a part of header translation that is transparent for either IPv4 node or IPv6 node in NAT-PT. Figure 5.2 presents the same example with Proxy Resolution.



1. ARP Request: who is 140.114.71.197
2. Neighbor Solicitation: who is 3FFE:3600::1
3. Neighbor Advertisement: IPv6 WWW Server is 3FFE:3600::1
4. ARP Reply: NAT-PT has 140.114.71.197
5. HTTP Request to 140.114.71.197
   SRC IP: 140.114.71.155
   DST IP:140.114.71.197
6. NAT-PT translate the HTTP Request to 3FFE:3600::1
   SRC IP: 3FFE:3600::140.114.71.155
   DST IP: 3FFE:3600::1
7. IPv6 WWW Server Reply the HTTP Request
   SRC IP: 3FFE:3600::21
   DST IP: 3FFE:3600::140.114.71.155
8. NAT-PT translate the HTTP reply to 140.114.71.155
   SRC IP: 140.114.71.197
   DST IP:140.114.71.155

Figure 5.2 Example of Transparent Resolution in NP-Based NAT-PT

- Pre-Resolution in Chassis-Based NAT-PT (Vitesse IQ2000)

With this manner of using Pre-Resolution for address resolution, the Chassis-Based NAT-PT collects the mappings of IP and MAC addresses of the nodes near the middle translator in two tables for coping with the layer 2 packets. Figure 5.3 illustrates an instance of connection establishment initiated from one IPv4 node to one IPv6 Web server.

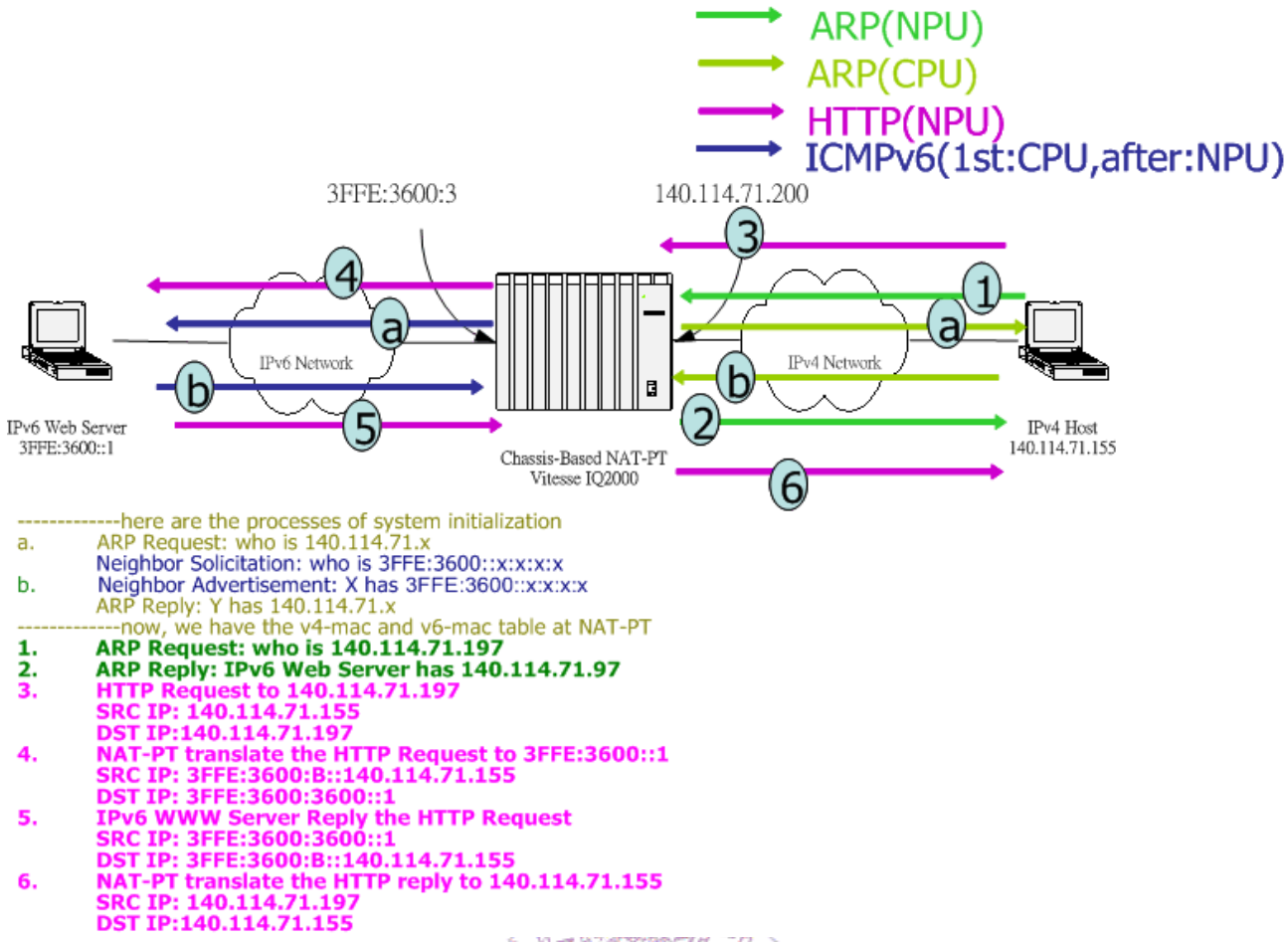


Figure 5.3 Example of Pre-Resolution in Chassis-Based NAT-PT

We have a summarization presented in Table 5.1 from the observation in Figure 5.1, Figure 5.2, and Figure 5.3.

Method	Platform	Connecting to the same Web Server	Connecting to different Web Server
Proxy Resolution	PC-Based	repeats step3 to step8 (6 steps)	repeats step1 to step8 (8 steps)
Transparent Resolution	NP-Based	repeats step5 to step8 (4 steps)	repeats step1 to step8 (8 steps)
Pre-Resolutiom	Chassis-Based	repeats step3 to step6 (4 steps)	repeats step1 to step6 (6 steps)

Table 5.1 Summarization of Three Resolution Methods

From the discussion above, we have a conclusion that the Chassis-Based NAT-PT using Pre-Resolution displays the least steps and the shortest time in building a HTTP connection. Maybe

there is no feeling for the end node to establish one HTTP connection to the same or different Web Server using these three methods, but it has conspicuous difference in building lots of HTTP connections among these resolving methods.

5.2. Performance Evaluation

We have performed a throughput test on our Chassis-Based NAT-PT system by connecting its IPv4 and IPv6 interfaces to the traffic generator like the Figure 5.4 showed. The traffic generator is configured to send a series of the same type of packets from one GbE port to the IPv4 or IPv6 interfaces with different sizes from 64 bytes to 1498 bytes in the direction of IPv4 to IPv6 and 84 4 bytes to 1518 bytes in the direction of IPv6 to IPv4, then capturing the packets which have been translated and transmitted by Chassis-Based NAT-PT device. Besides, we have an assumption in our throughput testing that the requiring tables such as IPv4-IPv6 address mapping table, IPv4-MAC table and IPv6-MAC table hav already constructed for the continuous data transferring.

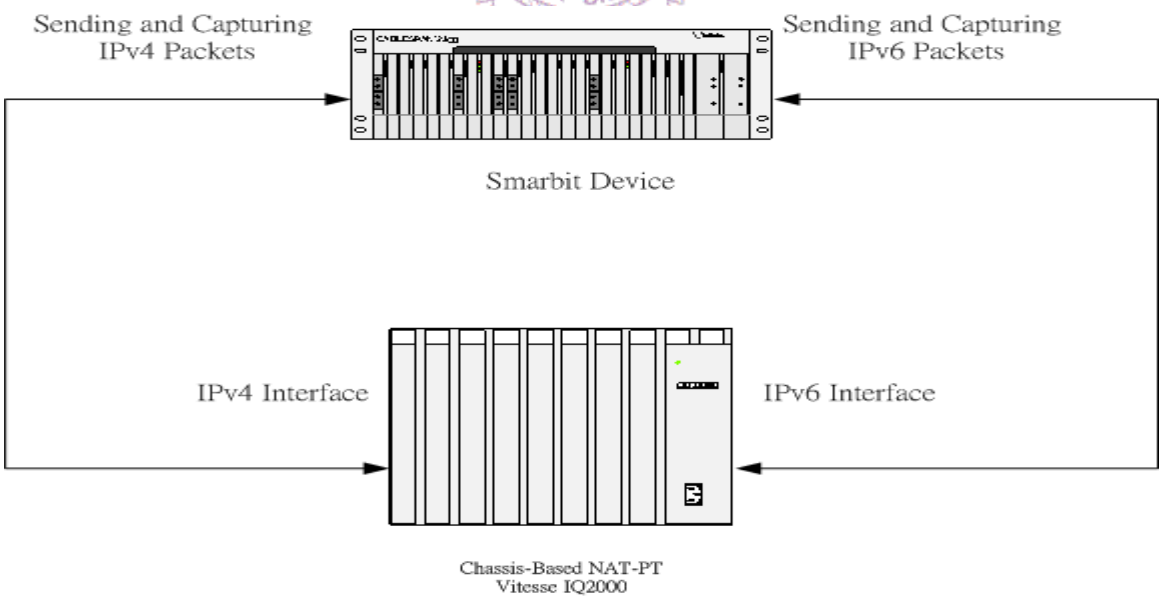


Figure 5.4 Throughput Testing Environment

We have accomplished the throughput test on our implementation in different direction. The throughput test from IPv4 to IPv6 is compared with that from IPv6 to IPv4. Figure 5.5 shows the

result of this comparison. As Figure 5.5 presented, we have found two things in the throughput test. First, except for 64-byte packet testing, our Chassis-NAT-PT implementation can achieve almost the same performance with that we perform the forwarding code (Input from PIM A and output from POM B without modifying the incoming packet) on the chassis platform. Our Chassis-NAT-PT can almost attain the wire speed when we give lots of packets with length larger than 128 bytes to saturate the GbE link. Second, the throughput of the direction from IPv4 to IPv6 is a little bit higher than the direction from IPv6 to IPv4 that has two explanations for this condition. One is relative with the limitation of the hardware architecture and one with the operation of NAT-PT defined in RFC. For the reason of hardware architecture in Vitesse IQ2000, since the issue of HEADER_SIZE discussed in chapter 4, the FACET has to move 32-byte “Payload” data from memory to Header Buffer in the packet flow from IPv6 while that is unnecessary from IPv4. On the other hand, in comparison with IPv4 packet the each IPv6 packet is 20-byte longer than IPv4.

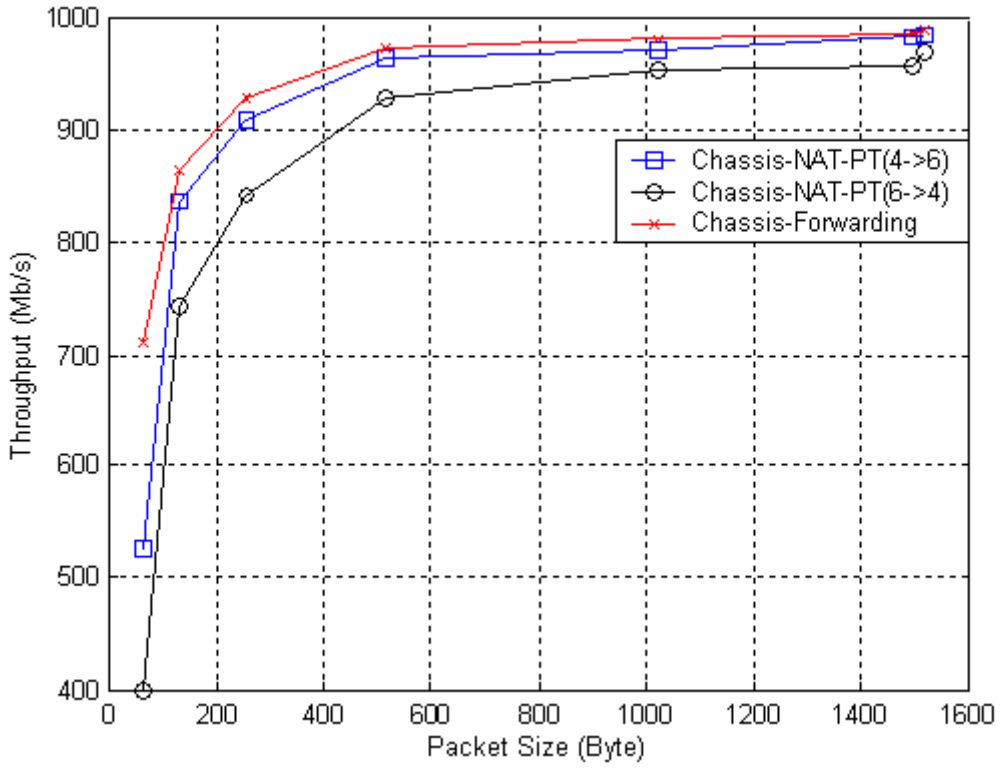


Figure 5.5 Throughput test of Chassis-Based NAT-PT