基于混合映射机制的 Napt-Pt 的研究与实现

陈 行 陶 军 吴 强
(东南大学计算机网络和信息集成教育部重点实验室 南京 210096)
(东南大学计算机科学与工程学院 南京 210096)

摘 要 IPv4 网络和 IPv6 网络之间的互联互通问题是 IPv6 网络研究中不可逾越的重要命题之一。在传统的 Nat-Pt 机制上引入端口映射机制,设计并实现了基于混合映射机制的翻译网关 Napt-Pt,对于 IPv4 端节点向 IPv6 端节点发起的连接采用地址映射机制,反之则采用端口映射机制。这使得翻译网关占用很少的 IPv4 临时映射地址就能应对大量的网络过渡负载,有效地缓解了网络过渡中的传输瓶颈。实验证明,采用混合映射机制的 Napt-Pt 翻译网关可以实现网络基本服务的过渡功能,而且对网络传输效率带来的影响非常有限,具有较高的实用性和可靠性。

关键词 IPv6,地址/协议转换技术,端口映射,网络过渡
中图法分类号 TP393.08 **文献标识码** A

Study and Implementation of Napt-Pt Based on Mix-mapping Mechanism

CHEN Hang TAO Jun WU Qiang
(Key Laboratory of Computer Network and Information Integration of Ministry of Education, Southeast University, Nanjing 210096, China)
(Department of Computer Science and Engineering, Southeast University, Nanjing 210096, China)

Abstract The problem of intercommunication between IPv4 and IPv6 network is a key issue that can not be ignored in IPv6 network research. Transition gateway Napt-Pt based on mixed-mapping mechanism was designed and implemented by adding port-mapping mechanism. If an IPv4 host launches a connection to an IPv6 host, address-mapping is used. If not, port-mapping is used. So transition gateway can use a small number of IPv4 temporary mapping addresses to deal with massive network transition payload, which mitigate the bottleneck in network transition. The experiment shows that transition gateway Napt-Pt based on mix-mapping mechanism can fulfill the transition of basic network service, and the influence on Network transmission efficiency is very limited. The approach is practicable and reliable.

Keywords IPv6, Napt-Pt, Port-mapping, Network transition

在过去的几十年中 IPv4 技术为互联网的起步和发展奠定了基础。但是随着计算机网络的普及,互联网的广泛接入,IPv4 地址的匮乏和路由表的过度膨胀问题日益突出。IETF 为此制定了下一代互联网互联协议 IPv6[1]。IPv6 把地址长度由原来的 32 位扩展成 128 位,有效解决地址资源不足的问题,同时在安全性、网络管理、移动性以及服务质量等方面也有明显的改进[2]。然而 IPv6 技术不可能在一夜之间部署完毕,IPv4 网络和 IPv6 网络必然有一段较长的共存期,必须提供一套完整的网络和应用过渡方案来解决 IPv4/IPv6 之间的互联互通问题。如何保证互联网服务在 IPv4 向 IPv6 的过渡时期中不被中断并保持良好的性能是下一代互联网发展和建设中首先需要解决的关键问题之一[3]。目前比较成熟的过渡方案技术主要分为 3 类:双协议栈技术[4]、隧道技术[4]和地址/协议转换技术 Nat-Pt[5]。其中双协议栈技术是各种过渡技术的基础,但是它要求每台主机都拥有全球唯一的 IPv4 和 IPv6 地址,并且要求网络设备同时支持 IPv4 和 IPv6 协议。IETF 提出的 Nat-Pt 技术只需要在 IPv4 网域和 IPv6 网域部署网络过渡网关就可以实现,而网络本身和端节点都无需修改,具有配置简单、透明度高、适应性广的特点。

通过对 Nat-Pt 技术的深入研究发现,IPv4 的动态地址映射能力是网络过渡中的瓶颈因素。为了节省稀缺的 IPv4 地址资源,增加 Nat-Pt 对 IPv4 和 IPv6 节点通信的支持能力,对 Nat-Pt 进行拓展,设计了基于端口映射机制的 Napt-Pt,一个 IPv4 地址可以映射多个网络连接,有效地解决 IPv4 临时地址短缺问题。在此基础上构建了一套网络基本服务过渡原型系统,为网络应用从 IPv4 到 IPv6 的平稳过渡奠定了坚实的基础。

1 Nat-Pt 过渡技术分析

1.1 Nat-Pt 翻译网关基本原理

Nat-Pt 系统由 3 部分组成:Nat,地址转换模块;Pt(协议转换模块);ALG(应用层网关)[6]。Nat 负责 IPv4 和 IPv6 地址之间的映射转换,配置一个 IPv4 的地址槽,将 IPv6 地址临时映射为全球可路由的 IPv4 地址,同时 Nat-Pt 网关还配置

到稿日期:2008-06-25 本文受国家自然科学基金重大研究计划项目(90604003)和国家自然科学基金项目(60603067)资助。
陈 行(1980—),男,博士研究生,主要研究方向为 IPv6 网络、网络安全,E-mail:chenhang@seu.edu.cn;陶 军(1975—),男,讲师,博士,主要研究方向为高性能网络、分布式计算、博弈与信息经济学;吴 强(1974—),男,博士研究生,主要研究方向为 IPv6 网络、无线 P2P 网络。

一段特殊 IPv6 地址 prefix，用于对 IPv4 地址的映射。Pt 部分采用 SIIT 技术[7]中的协议转换方法进行 IP4/IPv6 网络层报头的翻译，主要包括了 IPv4/IPv6 包头互译、ICMPv4/ICMPv6 报头互译和 ICMPv4/ ICMPv6 差错报文互译。ALG 模块负责对应用协议进行解释翻译，处理协议中包含 IP 地址的特殊应用，如 DNS 应用、FTP 应用等[8]。

DNS-ALG 模块是 Nat-Pt 网络基本服务过渡机制中的重要支撑。当 IPv4 端点发起会话访问 IPv6 网络时，首先通过访问 DNS 获得 IPv6 目的地址，DNS-ALG 从 IPv4 地址池中取得一个临时 IPv4 地址发送给 IPv4 端节点作为 IPv4 目的地址，并建立起临时 IPv4 地址和 IPv6 目的地址之间的映射关系。当 IPv6 节点发起会话访问 IPv4 网域的时候，先通过 DNS 服务获得 IPv4 目的地址。DNS-ALG 在所获得的 IPv4 目的地址前加上 IPv6 地址前缀 prefix，形成一个特殊的 IPv6 目的地址。这个特殊的 IPv6 地址包含 IPv4 地址，路由指向 Nat-Pt 网关。而源地址的地址变换关系是在会话的首个包到达 Nat-Pt 网关时进行。但是 IPv4 地址是有限的，IPv4 地址池中的临时地址个数成为 IPv4 端点和 IPv6 端点相互访问的瓶颈。

1.2 混合型 Napt-Pt 端口映射机制

为了节省稀缺的 IPv4 地址资源，增加 Nat-Pt 对 IPv4 和 IPv6 节点通信的支持能力，对 Nat-Pt 进行拓展，使用了端口和地址的组合替代单纯的映射地址，提高全局 IPv4 地址的使用效率。一个 IPv4 地址可以处理 63k 个 TCP 会话和 63k 个 UDP 会话，也就是一个 IPv4 地址就可以提供 63k 个可供利用的映射单元。这里需要考虑到，网络服务中的基本服务使用固定端口来作为服务类型的标志，例如 Web 服务默认使用 80 端口，FTP 服务默认适用 21 端口，DNS 服务使用 53 端口，作为网络中间设备的 Napt-Pt 不可能要求网络端节点改变这些端口使用原则。这意味翻译网关 Napt-Pt 不能修改网络连接中的目的端口，只能修改源端口。为此这里设计并实现了混合型 Napt-Pt，分别采用地址映射和端口映射两种机制，将地址池中的临时地址分成地址映射 IPv4 地址和端口映射 IPv4 地址两类。当网络会话是从 IPv4 方发起时，使用地址映射 IPv4 地址，将端口置为 0，使端口映射在报文翻译过程中不发生作用。当会话是从 IPv6 站点发起的时候，使用端口映射 IPv4 地址及其端口组合建立映射关系，提高 IPv4 地址的使用效率。

需要特别指出的是，Napt-Pt 中还拥有固定映射 IPv4 地址，用于建立 IPv4 地址和 IPv6 地址之间的静态地址绑定。这个静态地址绑定被用于向 IPv6DNS 服务器发出的查询报文翻译，它在 Napt-Pt 服务程序启动时就建立，并且不会被撤销。

2 基于 Napt-Pt 的过渡方案设计

2.1 地址映射数据结构

首先由从文件 IPv4_Addresses. list 中读取 IPv4 地址作为地址池中的地址。然后分别创建两条链表：链表 pStart_IPv4_List 存储独立映射 IPv4 地址，链表 pStart_PORT_IPv4_List 存储端口映射 IPv4 地址。链表节点的数据结构均为：

```
struct IPv4_Address_List
{
    struct in_addr
IPv4_Address;
    unsigned short Port;
    Boolean Free;
    struct IPv4_Address_List * pNext;
    };
```

其中保存独立映射 IPv4 地址的链表中的节点元素 IPv4_Address_List. Port 全部被赋为零。也就是说其中一个节点只保存了一个 IPv4 地址信息，只能对应一台 IPv6 主机地址。而保存端口映射 IPv4 地址的链表则会为每一个 IPv4 地址与其端口的组合设置一个节点。例如 202. 119. 11. 40 10001..10105 这个地址项会在链表中添加 105 个节点，这 105 个节点每一个都可以对应一个跨 IPv4-IPv6 网域的 TCP 连接，不论这些连接所牵涉到的 IPv6 主机地址有多少个。

为进行 IPv4-IPv6 的报文互译，需要为 IPv6 地址设置一个 IPv4 映射地址。为保存生命周期内的映射信息以供查询，这里又创建两条链表，分别为地址映射信息链表 pStart_Mapping_List和端口映射链表 pStart_PORT_Mapping_List。在地址映射机制中，一个 IPv4 映射地址对应一个 IPv6 的主机地址，根据翻译网关收到报文的目的地址和源地址在对应的地址映射链表中寻找对应的映射地址节点。在网络应用中，不同的连接用不同的地址及端口相区别，一个地址和端口的组合只能对应网络应用中的一次连接。因此在端口映射机制中需要根据报文的目的地址、源地址和目的端口、源端口来判断此报文对应于哪一个地址与端口的映射组合。链表节点结构为：

```
struct Mapping_List
{
      struct in_addr
IPv4_Address;
    unsigned short   IPv4_Port;
    struct in6_addr   IPv6_Address;
    unsigned short   IPv6_Port;
    int Number_of_Sessions;
    int TCP_Timer_Value;
    int UDP_Timer_Value;
    struct Mapping_List   * pNext;
};
```

2.2 地址映射机制的选择

对于由 IPv4 网域向 IPv6 网域发起的连接，报文映射应该使用地址映射机制。而对于由 IPv6 网域向 IPv4 网域发起的连接，报文映射应该使用端口映射机制。利用 Linux Socket Filter 可以从协议栈中获取收到的报文，将其导入到用户态。函数 NAT_Check_If_NATPT_Deal_IPv6()判断报文是应用地址映射机制还是应用端口机制。具体流程如图 1 所示。

当 Nat-Pt 主机收到的是 IPv6 报文，就应该利用源地址和源端口先后查询地址与端口的组合映射链表 pStart_PORT_Mapping _List 和地址映射链表 pStart_Mapping_List 中相对应的映射节点。如果在前者中查找到，就应使用端口映射机制，如果在后者中查找到，就应使用地址映射机制。如果都没有找到对应的映射表项，就认为这个报文是连接的发起报文，由此可判定这是由 IPv6 网域向 IPv4 网域发起的连接，应使用端口映射机制。从链表 pStart_PORT_IPv4_List 中取得一个 IPv4 地址端口组合，用这个组合与 IPv6 报文的源地址、

源端口一起生成一个新的映射节点，再将这个映射节点添加到端口映射链表 pStart_PORT_Mapping_List 的尾部。

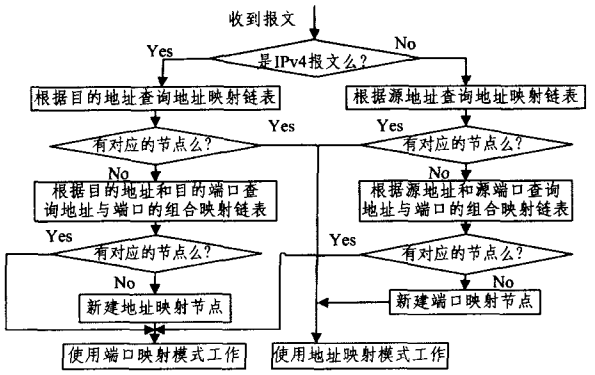

图 1　映射机制选择流程

对于接收到的 IPv4 报文，其处理方案与 IPv6 报文的处理方案相类似，只是在查询映射信息时使用的是 IPv4 报文的目的地址和目的端口。需要注意的是，IPv4 端节点只能利用域名来主动访问 IPv6 端节点，这意味着必需要 DNS-ALG 的支持。IPv4 节点只由 IPv4 网域内的 DNS 服务器通过翻译网关向 IPv6 网域内的 DNS 服务器发出查询请求。当 IPv6 网域中的 DNS 服务器向 IPv4 网域内的 DNS 服务器返回查询结果时，翻译网关将查询结果中的 IPv6 地址转换成 IPv4 地址，同时建立映射关系。首先根据 DNS 查询结果中的 IPv6 地址在地址映射链表 pStart_Mapping_List 中查找对应的节点，如果没有找到，则从链表 pStart_IPv4_List 中取得一个空闲的 IPv4 映射地址，将其和 DNS 查询结果中的 IPv6 地址组合在一起生成一个地址映射节点，并添加在地址映射链表 pStart_Mapping_List 的尾部。所以当 IPv4 端节点向 IPv6 端发起连接的时候，其第一个报文的目的地址肯定能在地址映射链表 pStart_Mapping_List 中查到。

2.3　报文应用层的翻译过程

如图 2 所示，通过对报文 Payload_type 的分析，确定报文是 TCP 还是 UDP 或 ICMP。通过报文端口判断此报文的应用层协议。如果有一个端口是 80，则是 http 协议；如果有端口是 21，则是 ftp 协议；如果有端口是 53，则是 DNS 的报文。对于 TCP 协议的报文，在地址翻译完成之后，根据报文长度变化，用 ALG_Manager_Retrieve _Sequence_Number_Offset ()函数调整 TCP 协议中的 sequence 字段。而对于 ftp 协议的报文，要根据 v4 版本的 ftp 协议和 v6 版本的扩展型 ftp 协议中传输控制命令的不同，对 ftp 的报文进行修改。然后要针对 ftp 协议中控制连接和数据传输连接的不同，对数据传输连接的地址和端口的组合映射节点的生命周期作特别延长。最后当然还要用 ALG_Manager _Calculate_Checksum_TCP()和 ALG_Manager_Calculate_Checksum_TCP _v6()重新计算报文的 checksum 字段。最后将报文重新写入目标网域的协议栈。

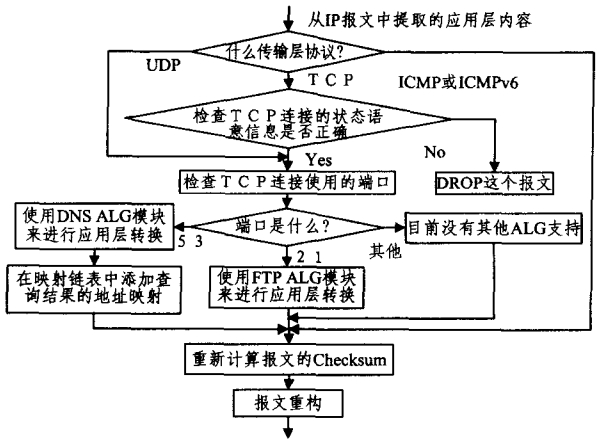

图 2　报文应用层翻译过程

3　实验环境和结果

3.1　网络拓扑和软硬件环境

为了充分检测各种基本网络应用的过渡效果，我们根据过渡技术的原型系统测试的实际需要，分别建立了 IPv4 网络和 IPv6 网络过渡测试环境，拓扑结构如图 3 所示。测试环境中配置有 Napt-Pt 翻译网关、IPv4 和 IPv6 DNS 服务器、FTP 服务器、Telnet 服务器以及 Web 客户端等设备，各设备通过 IPv6 交换机进行连接，并利用隧道机制通过一台华为路由器接入到中日 IPv6 网络中。服务器硬件均使用 CPU Intel P4 2.4G 内存 RAM 256M，网卡 Intel PRO/100M 的服务器，操作系统采用 RedHat Linux 8.0-2.14.18。IPv4 和 IPv6 DNS 域名服务软件使用 BIND-9.4；Web 服务软件采用 Apache-2.0.1；Email 服务软件 qmail；FTP 服务软件采用 Vsftpd-2.0.1；Telnet 服务软件采用 Telnetd-0.17。Web 客户端主机操作系统采用 WindowsXP Professional。

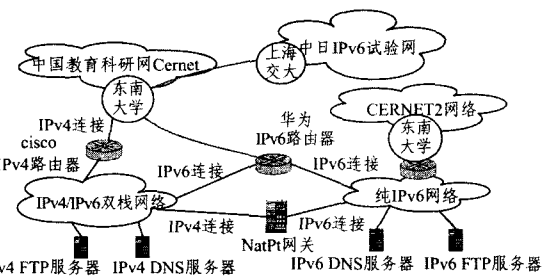

图 3　基本服务过渡测试拓扑

3.2　连通性能测试

这里同时在 IPv4 和 IPv6 网域的多台机器上用 ping 和 ping6 命令发出 ICMP 报文，各记录一百组数据，根据 ping 响应时间来研究 Napt-Pt 翻译网关的连通和网络延迟特性。这里用 IPv6_to_IPv4 来表示 IPv6 端节点向 IPv4 端节点发出的报文，用 IPv4_to_IPv6 来表示 IPv4 端节点向 IPv6 端节点发出的报文。

从表 1 中可以看出，从 IPv6 网域向 IPv4 网域发出的 ICMP 报文响应延迟稳定在 0.5ms 左右。延迟时间的方差也控制在很小的范围内。而 IPv6 网域内部的 ICMP 响应延迟稳定在 0.271ms 至 0.32ms 之间。Napt-Pt 翻译网关对响应延迟的影响稳定在 0.21ms 左右。

从表 2 可以看出，从 IPv4 网域向 IPv6 网域发出的 ICMP 报文响应延迟平均在 0.52ms，而且延迟数据的方差非常小，只有 0.0015。而 IPv4 网域内部的 ICMP 报文延迟在 0.27ms 至 0.284ms 之间。Napt-Pt 翻译网关对响应延迟的影响只有 0.24ms 左右。

反复观察 ICMP 响应延时抖动情况，我们发现 IPv6 端节点向 IPv4 端节点发送的第一个 ICMP 报文有特别长的响应

（下转第 64 页）

gateway[A]// Proc. of the IEEE INFOCOM[C]. New York: IEEE Press, 1999: 1320-1328

[4] Ott TJ, Lakshman TV, Wong LH. SRED: Stabilized RED[A]// Proc. of the IEEE INFOCOM[C]. New York: IEEE Press, 1999: 1346-1355

[5] Liu S, Basar T, Srikant R. Exponential-RED: A stabilizing AQM scheme for low-and high-speed TCP protocols[J]. IEEE/ACM Trans. on Networking, 2005, 13(5): 1068-1081

[6] Hollot CV, Misra V, Towsley D, et al. On designing improved controllers for AQM routers supporting TCP flows[A]// Proc. of the IEEE INFOCOM[C]. Anchorage: IEEE Press, 2001: 1726-1734

[7] 卢锡城，张明杰，朱培栋. 自适应 PI 主动队列管理算法[J]. 软件学报，2005，16(5)：903-910

[8] Zhang HY, Liu BH, Dou WH. Design of a robust active queue management algorithm based on feedback compensation[A]// Proc. of the ACM SIGCOMM[C]. Karlsruhe: ACM Press, 2003: 277-285

[9] 王秀利，王永吉，周辉，等. 基于 D 稳定域和 ITAE 准则的主动队列管理算法[J]. 软件学报，2007，18(12)：3092-3103

[10] Wang X L, Wang Y J, Zhou H, et al. PSO-PID: a novel controller for AQM routers[A]// Proc. of the IEEE/IFIP WOCN[C]. Bangalore: IEEE Press, 2006: 1-5

[11] 王秀利，王永吉. 一种开放源代码的网络仿真器的原理与实现[J]. 计算机工程与应用，2004，40(15)：137-140

[12] 金信苗. 基于 ns2 的 LEO 卫星网络路由算法模拟[J]. 计算机科学，2007，34(1)：57-60

[13] Nsnam[EB/OL]. http://www.isi.edu/nsnam/

[14] Fall K, Varadhan K. The ns Manual[EB/OL]. The VINT Project: UC Berkeley, LBL, USC/ISI, and Xerox PARC. 2003

(上接第 35 页)

延迟时间，达到 1ms 左右。而 IPv4 端节点向 IPv6DNS 服务器发出的 ICMP 报文延迟则没有这种现象。这种区别是由 Napt-Pt 翻译网关的内部工作机制带来的。前者翻译 ICMP 报文时使用了端口映射机制，后者采用了 3.2 节描述静态地址绑定，其地址映射被固定在地址映射链表中，不需要获取新的地址映射 IPv4 地址，并对映射链表进行复杂的插入和删除。

这里特别指出，ICMP 报文中没有端口字段，但是有 Identifier 字段用以标示不同的 ping 进程。这里用 ICMP 报文中 Identifier 值作为映射机制中的目的端口和源端口。

表 1 IPv6_to_IPv4 的 ICMP 报文响应延迟时间

(单位毫秒)	最小时间	平均时间	最大时间	时间方差
穿越 Napt-Pt 的应答延迟	0.449	0.493	1.06	0.0136
纯 IPv6 环境下的应答延迟	0.269	0.2812	0.318	0.00012

表 2 IPv4_to_IPv6 的 ICMP 报文响应延迟时间

(单位毫秒)	最小时间	平均时间	最大时间	时间方差
穿越 Napt-Pt 的应答延迟	0.474	0.5256	0.650	0.0017
纯 IPv4 环境下的应答延迟	0.268	0.2795	0.282	0.0000129

3.3 FTP 传输性能分析

表 3 中用 IPv6_to_IPv4 来表示 IPv6 端节点向 IPv4 端节点发出的连接，用 IPv4_to_IPv6 来表示 IPv4 端节点向 IPv6 端节点发出的连接。我们发现经过 Napt-Pt 过渡的 FTP 连接响应时间和传输速度比纯 IPv4 和纯 IPv6 以太网环境下略有下降。这是因为 Napt-Pt 本身对分组的转换过程需要一定的时间，而且 Napt-Pt 工作于网络层，FTP_ALG 更是处在应用层，每个报文的翻译都要经过翻译网关从底层到高层的多层处理，造成时间上面的损耗。我们还发现 Napt-Pt 技术的双向连接性能存在一些差异，IPv6_to_IPv4 情况下 FTP 建立连接响应时间和文件传输速度稍慢于 IPv4_to_IPv6。由图 3 可知，程序首先查找地址映射链表，然后再查找端口映射链表，由结果来判断报文采用的映射机制。IPv4_to_IPv6 情况下采用的是地址映射机制，会直接从地址映射链表中得到查询结果，从而跳过了端口映射链表的查询，这使得 IPv4_to_IPv6 时查找链表的时间耗费更低，从而有更快的报文翻译速度。从表 3 中可以看到，经过翻译网关的网络传输速度与单纯网络环境下的传输速度的差距很小，翻译网关使用 2.8G 的 CPU 已经基本满足百兆网络满负荷传输的需要了。

这里还要特别指出，无论 FTP 协议采用何种工作机制，都有两条 TCP 连接，控制连接和传输连接。端口映射机制下会为这两条 TCP 连接分别建立两个映射节点。当传输连接对应的映射节点还存在的时候，其控制连接对应的映射节点绝不能因为长时间没有报文传输而被清除。也就是说与 FTP 传输相关的这两个端口映射节点的生命周期必须被同步更新。

表 3 FTP 传输性能比较

(单位毫秒)	以太网		基于 Napt-Pt 技术	
	纯 IPv4 环境	纯 IPv6 环境	IPv4_to_IPv6	IPv6_to_IPv4
连接响应时间	4.923ms	4.893ms	5.423ms	5.522ms
FTP 传输速度	11.1MB/s	11.2MB/s	11.0MB/s	10.0MB/s

结束语 本文在 Nat-Pt 动态地址映射的研究基础上添加了端口映射机制，实现了基于混合映射机制的 Napt-Pt 翻译网关。根据网络连接发起的方向不同，分别采用地址映射机制和端口映射机制，使用少量 IPv4 地址就能处理大量会话，扩展了翻译网关的传输能力。结合项目需求搭建了完善的实验环境。实验证明，Napt-Pt 翻译网关技术能有效地实现网络基本服务的过渡功能，而且对网络传输的影响非常有限。

本文结合项目要求实现了网络基本服务的过渡，然而网络服务的种类非常丰富，未来将对 Napt-Pt 翻译网关进行更多的扩展，并运用更多种类的网络服务测试来验证翻译网关的可靠性和扩展性。

参 考 文 献

[1] Bradner S, Mankin A. RFC 1752. The recommendation for the IP next generation protocol [S]. IETF, 1995

[2] Deering S, Hinden R. RFC 2460. Internet Protocol, Version 6 (IPv6) Specification [S]. IETF, 1998

[3] Davies J. Understanding IPv6. Second Edition [M]. Seattle: Microsoft Press, 2002: 76-83

[4] Gilligan R, Nordmark E. RFC 2893. Transition Mechanisms for IPv6 Hosts and Routers [S]. IETF, 2000

[5] Tsirtsis G, Srisuresh P. RFC 2766. Network Address Translation-Protocol Translation (NAT-PT) [S]. IETF, 2000

[6] Nat-PtImplementationIntro[OL]. http://www.eurescom.ed/~public-webspcae/P1000-series/P1009/ doc3_1.html

[7] Nordmark E. RFC 2765. Stateless IP/ICMP Translator (SIIT) [S]. IETF, 2000

[8] [Allman M, Ostermann S. RFC 2428. FTP Extensions for IPv6 and NATs [S]. IETF, 1998