Mobility in IPv4-IPv6 Transition Scenarios

Luís Oliveira [1], António Amaral [2], Amaro de Sousa [3]
[1] Institute of Telecommunications / Polytechnic Institute of Tomar, Portugal
[2] Institute of Telecommunications – pole of Aveiro, Portugal
[3] Institute of Telecommunications / University of Aveiro, Portugal

Abstract

This paper discusses the provision of mobile IP services on IPv4-IPv6 transition scenarios. There is no single IPv4-IPv6 transition mechanism that can solve this problem. We present solutions and discuss their limitations based on the combination of different transition mechanisms that were experimentally validated.

I. INTRODUCTION

Although there are already solutions to provide mobile IP in IPv4 networks and in IPv6 networks, the provision of mobility in mixed networks has yet many uncertainties. It is known that the evolution of the current Internet, based on IPv4 protocol, to the future IPv6 Internet will be based on transition scenarios. In this migrating path, there is the need to introduce as soon as possible new network services that are considered of key importance for the objectives of the future IPv6 Internet. One of these services is mobility.

Currently, there are several transition mechanisms proposed by IETF to solve different aspects on the provision of point-to-point IP connectivity in IPv4-IPv6 transition scenarios. However, none of these mechanisms was aimed to address the specific needs of mobile IP. In fact, there is no single IPv4-IPv6 transition mechanism that can provide IPv4-IPv6 connectivity to mobile nodes.

This paper addresses the issue on how to combine the different transition mechanisms in order to provide seamless mobile IP service. Mobility is addressed in this work in the perspective of a nomadic user that requires to initiate sessions to any other host with its home address and requires that other hosts initiate sessions with him based also on its home address. We address the mobile service provision both to IPv4 and IPv6 hosts. We did not consider the situation of a dual stack host with mobile IP service because, at the time of this work, there was no available implementations that implement this feature.

This paper is organized as follows: sections II and III briefly review IPv4-IPv6 transition mechanisms and mobile for both protocol stacks; section IV discusses how to provide mobility to IPv6 hosts and section V discusses the IPv4 case.

II. IPV4-IPV6 TRANSITION MECHANISMS

IPv4-IPv6 transition mechanisms can be classified in three different types: dual stack, tunneling and translation [1].

In dual stack mechanisms, IPv4 and IPv6 protocol stacks are both supported in network devices (routers) and it is the most straightforward way to implement transition scenarios. Hosts with the same protocol stack can communicate with each other and dual stack hosts (hosts with both protocol stacks) enable applications to choose the appropriate protocol stack. However, this mechanism requires a double effort to run both protocol stacks on network elements and it does not solve the inter-working between IPv4 hosts and IPv6 hosts.

Tunneling mechanisms assume that end systems (either networks or hosts) have the same IP version protocol stack but intermediate networks support only the other IP version. Network elements that connect networks of different IP versions must be dual stack and these elements are the end-points of tunnels. Tunnels are implemented through encapsulation: the tunnel entry node puts each original IP packet in the payload of an IP packet of the other version and sends it to the tunnel exit node address; the exit node does the inverse operation. Besides manual configuration of tunnels, different tunneling mechanisms were proposed in IETF (Internet Engineering Task Force) to provide IPv6 connectivity over IPv4 tunnels, e.g., 6over4, 6to4, and DSTM.

Dual stack and tunneling mechanisms are useful for many transition scenarios on interest but, like dual stack mechanisms, do not enable the inter-working between IPv6 and IPv4 end systems. To perform this interaction, translation mechanisms are required. The translation task is done either at the IP layer involving translation of IP addresses and mapping of IP header fields (e.g., BIS, NAT-PT) or at higher layers (e.g., TRT, SOCKS64).

The rest of this section reviews the IPv4-IPv6 transition mechanisms that were considered in our work:

Configured IP-in-IP tunnels: The "tunnel" nodes are statically configured to perform tunneling. The tunneling parameters are managed either manually or via some automated service provided by a tunnel broker. The tunnel broker is a network element that, when requested by the routers, returns the appropriated tunnel configuration scripts and parameters [2].

6to4: This mechanism [3] is an automatic router-to-router IPv6 over IPv4 tunnel that uses the IANA assigned IPv6 TLA prefix 2002:/16 to identify a 6to4 site followed by the IPv4 address (2002:V4ADDR::/48) of this site. With this mechanism, the tunnel entry node extracts the IPv4 address of the exit node through the prefix of the IPv6 destination address and, thus, is able to perform automatic tunneling. The connectivity between a 6to4 IPv6 network and a native IPv6 network is possible via a 6to4 relay router. This router must have at least one logical 6to4 interface and at least one IPv6 interface.

NAT-PT: The Network Address Translation - Protocol Translation [4] is a statefull IPv4/IPv6 translator that uses the SIIT algorithm [5] to perform the translation between IPv4 and IPv6 packet headers. It runs on a dual stack network element that acts as a default router between the IPv4 domain and the IPv6 domain. From a conceptual point of view, NAT-PT can be seen as an extension of the conventional IPv4 NAT mechanism. One problem that NAT-PT faces is that many applications (e.g. FTP, DNS) have embedded IP addresses. The support of these protocols requires Application Layer Gateway (ALG) modules that can reflect the IPv4-IPv6 address translation operation at the applications layers.

TRT : The Transport Relay Translator [6] enables direct communication between IPv6 only and IPv4 only hosts. It does the translation at the transport layer and, like NAT-PT, it runs on a dual stack network element that acts as a default router between the IPv4 domain and the IPv6 domain. As defined in the current RFC, sessions are initiated only from IPv6 side. The IPv6 address of the IPv4 host (assigned to identify it in the IPv6 domain) is composed by a general 64 bit prefix followed by the IPv4 address of the destination node and the TRT node has to be the default router to this virtual IPv6 network. From a conceptual point of view, TRT acts as a proxy server that accepts connections from IPv6 hosts in the name of IPv4 hosts and establishes connections to IPv4 hosts in the name of IPv6 hosts. Sessions initiated from IPv4 side are not defined in RFC and are usually not supported by current implementations since they require manual IPv4-IPv6 address mapping and no automatic procedure is yet available (it is possible to embed an IPv4 address into an IPv6 address but it is not possible to do the inverse operation).

III. MOBILE IP

Mobile IP enables a Mobile Node (MN) to maintain the same address in all communications with any Correspondent Node (CN) while moving from one network to another.

Mobile IPv4 [7] defines two entities to provide mobility support: a Home Agent (HA) and a Foreign Agent (FA). The HA is statically assigned to the MN based on its permanent IPv4 Home Address. The FA is assigned to the MN based on its current location. The FA has an associated IP address called care-of address. When MN is outside its home network, packets destined to him are intercepted by its HA, and tunneled to the FA using the care-of address as the exit tunnel address. The FA decapsulates the packets and forwards them directly to MN. In the opposite direction, MN sends all packets directly to the CN using its Home Address as source address (triangular routing concept).

The existence of a FA is not strictly required and, as an alternative, MN may use a co-located care-of address that can be obtained via some dynamic assignment protocol such as DHCP. In this case, MN performs the decapsulation of packets sent by HA. Nevertheless, this solution requires more IPv4 addresses to be available on foreign networks which is usually the main reason for the use of the FA entity.

Mobile IPv6 [8] has some improvements over mobile IPv4. First, due to the large IPv6 address space and the IPv6 address auto configuration mechanism, the FA entity is no longer required. Second, MN uses the Binding Mechanism on the destination options of IPv6 header to announce to CN its current care-of address; in this way, the first packet from CN to MN is the only one to go through the HA (this avoids the triangle routing problem of mobile IPv4) and the others flow directly between CN and MN.

Recent implementations of mobile IP for both versions support reverse tunneling. In this case, MN sends IP packets through a reverse tunnel to its HA that decapsulates the original packets and send them to CN. Although it seems a routing non-optimized approach, it is useful to make mobile IP compatible with other network operation requirements.

IV. TRANSITION SCENARIOS WITH MOBILE IPv6

The provision of mobile service to IPv6 hosts was investigated in three different transition scenarios that are illustrated in Figure 1. The differences rely on the type of IP protocol supported in the Intermediate Network and in the Correspondent Network (the network that hosts the CN) and are presented in Table 1 (the fourth possible scenario corresponds to the pure mobile IPv6 service provision).

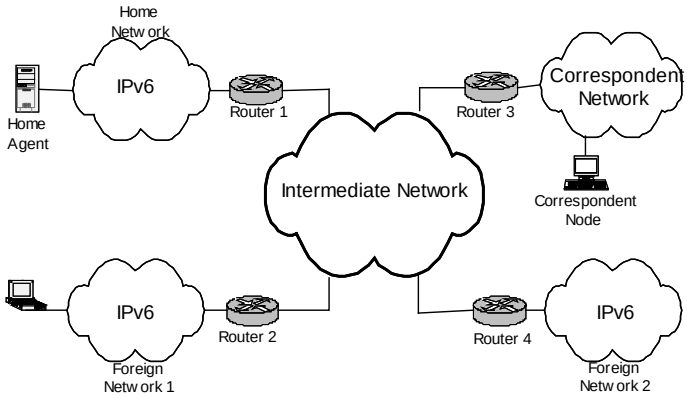

Fig. 1- Transition scenarios specification for mobile IPv6.

Table I

	Intermediate Network	Correspondent Network
Scenario 1	IPv6	IPv4
Scenario 2	IPv4	IPv6
Scenario 3	IPv4	IPv4

IPv6 mobile elements MN and HA were based on MIPL release 9.4 software [9] with Red HAT 7.3 Linux Kernel release 2.4.18. At the time of this work, this implementation was compliant with Mobile IPv6 draft 15 and included reverse tunneling (this feature is important to solve some operational issues as will be understood later).

A. Scenario 1

This scenario considers three IPv6 routers (Router 1, Router 2 and Router 4) and one dual stack router (Router 3). A translation mechanism is required on Router 3 since MN and CN run different IP protocol stacks. We have investigated two solutions. The first one is to use the TRT mechanism on Router 3 and all routers are FreeBSD based elements with the KAME snap kit [10] (which includes the TRT mechanism). This solution requires little configuration effort and runs successfully for sessions initiated by MN to CN but, as explained in the TRT description, does not enable sessions to be initiated by CN.

To avoid this limitation, a second solution was investigated based on the NAT-PT mechanism running on Router 3 and the use of reverse tunneling on mobile IPv6. Reverse tunneling is required because NAT-PT cannot process the IPv6 destination options where MN home address are carried. Using reverse tunneling, packets sent for CN are sent through HA with the MN Home Address as their origin IP address. Therefore, packets from MN to CN are received by NAT-PT (in its IPv6 interface) always with the MN Home Address as the packet source address. In this solution, the NAT-PT Router 3 was based on Windows 2000 OS with Service Pack 1 running NAT-PT Microsoft implementation and all other routers were the same as in the previous solution.

The two solutions are equivalent in terms of configuration effort. In terms of performance, the second solution provides mobile service for sessions initiated in both ways but it requires additional processing on HA entity (it must relay all packets between MN and CN) and penalizes the resource utilization of Intermediate Networks (since IP flows go through HA in both ways).

B. Scenario 2

This scenario that can be implemented through tunneling mechanisms since MN and CN run the same IPv6 protocol stack. Both configured tunnels and 6to4 mechanism are possible. The second case requires that the IPv6 networks be identified by the 6to4 prefix, while the configured tunnels give complete freedom in the choice of IPv6 network addresses. Note that to use the 6to4 transition mechanism in an IPv6 network, we must assign the prefix 2002::/16 followed by the address of the IPv4 interface of its gateway router. All routers were FreeBSD based with the KAME snap kit (which includes configured tunnels and 6to4 mechanism) and in both cases were run successfully.

In operational terms, the 6to4 mechanism is more scalable than configured tunnels since there is no need for any configuration action on current 6to4 routers when new 6to4 networks are connected to the Intermediate Network.

C Scenario 3

Like scenario 1, this scenario requires a translation mechanism since MN and CN run different IP protocol stacks. The most straightforward place to run the translation mechanism is on Router 1 since it is the Home Network gateway. We have investigated two solutions. The first one is based on the use of the TRT mechanism running on Router 1 combined with the use of IPv6 tunneling over IPv4 between the Home Network and all other IPv6 networks. For the tunneling part of this solution, both configured tunnels and 6to4 were successfully tested. All routers were FreeBSD based with the KAME snap kit. Similar to scenario 1, this solution does not enable sessions to be initiated by CN.

A second solution was implemented based on the use of NAT-PT mechanism running on Router 1 combined with the use of reverse tunneling on mobile IPv6 (due to the inability of NAT-PT to deal with the destination options) and with the use of IPv6 tunneling over IPv4 between the Home Network and all other IPv6 networks. Once again, for the tunneling part of this solution, both configured tunnels and 6to4 were successfully tested. The NAT-PT router 1 was based on Windows 2000 OS with Service Pack 1 running NAT-PT Microsoft implementation and all other routers were the same as in the previous solution.

Although the second solution enables sessions to be initiated in both directions, it is not a scalable solution for many reasons: (i) it requires additional processing on HA, (ii) it requires more management effort if 6to4 mechanism cannot be used and (iii) it relies on tunnels over tunnels with high bandwidth penalty because of cumulative packet overhead on switching performances of network nodes.

V. TRANSITION SCENARIOS WITH MOBILE IPv4

The provision of mobile service to IPv4 hosts was investigated in three different transition scenarios that are illustrated in Figure 2. Similar to the previous section, the differences rely on the type of IP protocol supported in the Intermediate Network and in the Correspondent Network (the network that hosts the CN) and are presented in Table 2 (the fourth possible scenario corresponds to the pure mobile IPv4 service provision).

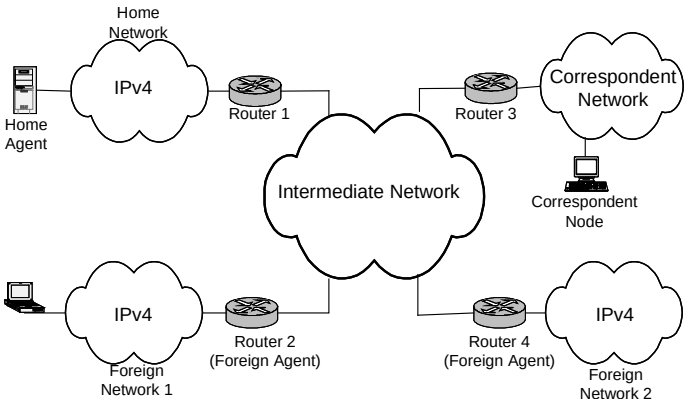

Fig. 2- Transition scenarios specification for mobile IPv4.

Table II

	Intermediate Network	Correspondent Network
Scenario 4	IPv4	IPv6
Scenario 5	IPv6	IPv4
Scenario 6	IPv6	IPv6

IPv4 mobile elements MN, HA and FA were based on Dynamics release 0.8.1 software [11] with Red HAT 7.3 Linux Kernel release 2.4.18. In all scenarios, Router 2 and Router 4 were based on this platform with the FA module activated.

A. Scenario 4

This scenario requires a translation mechanism and is efficiently solved through the use of NAT-PT mechanism on Router 3. It is compatible with mobile IPv4 because in this version, the MN uses its home address (and not its care-of address as in mobile IPv6) as the origin IP address of sending packets. In this case, Router 1 was based on FreeBSD with the KAME snap kit and Router 3 was based on Windows 2000 OS with Service Pack 1 running NAT-PT Microsoft implementation.

Note that in this case, the TRT mechanism running on Router 3 is not considered as an alternative solution since there is no advantage of using it: (i) it only supports the initiation of sessions from CN to MN which is usually not the case and (ii) TRT mechanism is more processor demanding than NAT-PT since it works at transport layer.

B. Scenario 5

This scenario requires tunneling mechanisms to support IPv4 connections over the IPv6 Intermediate Network. The only tunneling mechanism available both on FreeBSD and on Linux based routers was configured tunnels. In this case, both Router 1 and Router 3 were FreeBSD based routers.

C. Scenario 6

This scenario requires a translation mechanism and, like in scenario 3, the most straightforward place to run it is on Router 1 since it is the Home Network gateway. The NAT-PT mechanism is the most appropriate solution (because of the reasons pointed out in scenario 4) but it requires the use of IPv4 tunneling over IPv6 between the Home Network and all other IPv4 networks. For the tunneling part of this solution, configured tunnels were used as they are the only available solution.

VI. CONCLUSIONS AND FURTHER WORK

In this work, we have presented experimental validation of network solutions that can support mobility to both IPv4 hosts and IPv6 hosts in network scenarios where IPv4-IPv6 connectivity is required. We showed that with appropriate combinations of configured tunnels, 6to4, NAT-PT and TRT transition mechanisms, it is possible to provide the mobile service in a wide range of transition scenarios. The mobile service is addressed in the perspective of a nomadic user that requires being able to communicate with other hosts with its home address independently of the current network location. The proposed solutions are, then, appropriate to data communications, which is, up to now, the dominant traffic supported by the present Internet. Mobility support of real-time communications has additional constraints (e.g., short service failures while moving from one network to another) that were not addressed in this paper. The experimental work reported in here is based on available software packages that run on open platforms and was implemented on the Networks Laboratory of Institute of Telecommunications – pole of Aveiro.

Currently, this work is being extended to study and integrate the DNS service on these scenarios, which is another important building block towards the effective evolution to the future IPv6 networks. In an integrated IPv4-IPv6 network, all hosts must have at least an IPv4 address and an IPv6 address under the same DNS name, which must be carefully coordinated with address translation mechanisms like NAT-PT and TRT.

ACKNOWLEDGEMENTS

The authors wish to thank PT Inovação for its financial support of part of the activities presented here.

REFERENCES

[1] Waddington D. G., Chang F., “Realizing the Transition to IPv6”, *IEEE Comm. Mag.*, June 2002

[2] Durand, A. et al., “IPv6 Tunnel Broker”, *IETF RFC 3053*, January 2001

[3] Carpenter, B., Moore, K., “Connection of IPv6 Domains via IPv4 Clouds without Explicit Tunnels”, *IETF RFC 3056*, February 2001

[4] Tsirtsis, G., Srisuresh, P., “Network Address Translation - Protocol Translation”, *IETF RFC 2766*, February 2000

[5] Nordmark E., “Stateless IP / ICMP Translation Algorithm (SIIT)”, *IETF RFC 2765*, February 2000

[6] Hagino, J., Yamamoto, K., “An IPv6-to-IPv4 Transport Relay Translator”, *IETF RFC 3142*, April 2001

[7] Perkins C., “IP Mobility Support”, *IETF RFC 2002*, October 1996

[8] Johnson D., Perkins C. Arkko J., “Mobility Support in IPv6”, *IETF draft-ietf-mobileip-ipv6-20.txt*, January 2003

[9] http://www.mipl.mediapoli.com, MIPL Mobile IPv6 for Linux

[10] http://www.kame.net, KAME Project

[11] http://www.cs.hut.fi/research/dynamics, Dynamics HUT Mobile IP