Integrated Routing and Addressing for Improved IPv4 and IPv6 Coexistence

Arturo Azcorra, *Senior Member, IEEE,* Michal Kryczka, *Student Member, IEEE*, and Alberto García-Martínez

***Abstract*—We claim that the slow deployment of IPv6 seen so far is related with the complete decoupling with IPv4 regarding to routing and addressing. In this paper we propose Integrated Routing and Addressing (InRA), a new solution to allow IPv6 to integrate with IPv4 routing and addressing in order to decrease management cost during the coexistence period, and speed up IPv6 deployment. Addressing and routing integration are achieved by the use of a new address format derived from IPv4 address assignment, and the use of a new type of encapsulation allowing packets to be forwarded according to the most detailed information known by a router (InRA or IPv4).**

***Index Terms*—IPv4, IPv6, integrated routing and addressing.**

I. Introduction

EXHAUSTION of IPv4 is steadily approaching, according to data provided by IANA and the Regional Internet Registries [1]. However, while IPv6 has been promoted for some time as the best solution to the upcoming depletion of IPv4 addresses, it has not yet succeeded in become widely deployed. In November 2008, on the occasion of 10th anniversary of the IPv6 standardization, Google has conducted a research to determine the state of IPv6 deployment [2]. The results indicated that the percentage of IPv6 prevalence in overall Internet traffic is less than 1% in each country, with Russia as a leader (0.76%). These numbers show clearly that even if IPv6 is a stable and well tested technology, its presence in the Internet is still very low, and concerns arise about the time required for a significant presence in the Internet. As a result, connectivity and further growth of the Internet could be compromised.

Several methods have been proposed to ease IPv6 deployment and assure IPv4 and IPv6 interoperability. Dual-stack networks, operating systems and applications using public IPv6 and private (NATed) IPv4 addressing behind a reduced number of IPv4 public addresses, is the recommended mechanism. For connecting IPv6 sites or hosts across IPv4 networks which cannot be easily upgraded, configured IPv6-in-IPv4 tunnels [3], IPv6 over MPLS tunnelling, or Teredo [4] can be used. If the applications or end sites are not migrated, then translation can be performed by the combination of NAT64 [5] for address translation and SIIT [6] for translating the rest of the packet fields, or by means of IVI [7]. All these mechanisms suffer from limitations, inefficiencies or security issues [8]. Dual-stack requires a duplicate effort in managing addressing and routing. Tunnels usually suffer from lack of optimality, and single points of failure. Finally, translation techniques usually restrict the type of packets and fields of the packets which can be translated.

It must be noted that these mechanisms were not designed to provide efficient coexistence with IPv4, but to serve as a disposable tool during the expectedly short transition to IPv6. However, this fast transition has not occurred and IPv6 will need long time to overtake IPv4 deployment even in the most optimistic forecasts. As a result of this scenario, some problems related with the coexistence of IPv4 and IPv6 that were largely underestimated must be revisited now. Part of these problems come from the lack of relation between IPv6 and IPv4 addressing, which has resulted in duplicating the cost to configure applications, hosts, middleboxes, etc. for stub sites, and to manage separately routing and its policies for both transit and stub sites. This observation is supported by our experience managing REDIMadrid, the research network of the region of Madrid, from 2002 to 2007, as a dual-stack network. Along with managing two different routing tables and policies, troubleshooting is also harder than before.

In order to reduce the burden in deploying IPv6, we propose a solution called Integrated Routing and Addressing (InRA). With this solution, IPv4 is not abandoned, but extended. IPv4 addresses and routing information are included into IPv6 addressing and routing, instead of considering an orthogonal deployment of IPv4 and IPv6. To do so, a new address type is defined and the whole routing information of IPv4 is included in an extended routing information table, also including routes specific to the new addressing model. InRA packets contain both IPv4 and InRA addresses, so forwarding can be performed according to the most precise information available. The integration between IPv4 and IPv6 is expected to simplify their combined operation, significantly reducing the *opex* of introducing and maintaining IPv6.

Compared to existing proposals aiming to support coexistence, 6to4 [9] shares some features with InRA, in the sense that they both provide means to traverse IPv4 infrastructure. However, 6to4 does so by keeping separated the addressing domains in the migrated networks, and does not integrate routing behaviour which, from our point of view, results in increased costs. Teredo [4] also uses tunnels, but in this case the aim is to allow a host inside a legacy IPv4 domain, in particular behind a tunnel, to establish communications with IPv6 hosts. InRA does not try to solve the connectivity problem without coordination with the network administrator, and does not require the use of servers, as it is the case for Teredo.

Manuscript received December 21, 2009. The associate editor coordinating the review of this letter and approving it for publication was G. Lazarou.

A. Azcorra and A. García-Martínez are with the Department of Telematic Engineering, University Carlos III Madrid, Av. Universidad 30, 28911 Leganes, Madrid, Spain (e-mail: {azcorra, alberto}@it.uc3m.es).

M. Kryczka is with IMDEA Networks, Av. del Mar Mediterraneo 22, 28918 Leganes, Madrid, Spain (e-mail: michal.kryczka@imdea.org).

Digital Object Identifier 10.1109/LCOMM.2010.05.092449

Higher 64 bits			Lower 64 bits
Prefix	IPv4 gateway address	24 bits	64 bits
40	a.b.c.d	Subnet ID	End host identification

Fig. 1. Structure of an InRA IPv6 address.

II. Integrating Addressing and Routing

The Integrated Addressing model proposed in InRA is based on the fundamental assumption that the extension of the unicast address space should be hierarchically dependent on the IPv4 existing one. Note that this is a very different approach to the conventional IPv6 unicast address space model, which has been defined as completely unrelated to the IPv4 address space.

To ensure compatibility with existing IPv6, we propose the use of a 128-bit address space for the extended address space (see Fig. 1). In particular, we suggest to reserve an unallocated IPv6 address space block for deploying the InRA address space (for example, 40::/8). We will use the term IPv6* to refer to this address space. Compatibility with IPv6 assures that existing applications can seamlessly run on top of the InRA architecture, and that the plethora of mechanisms adapted to IPv6 (DNS, DHCP, neighbour discovery, etc.) can be used.

To build an InRA unicast prefix, the 32 bits corresponding to a unicast public IPv4 address are included after the first 8 bits identifying the InRA address space. We call this 32 bits the "v4* prefix" of the IPv6* address. In this way, a topological relationship between the IPv4 and the IPv6* address space is established. All the IPv6* addresses sharing a given v4* prefix must be in the topological "vicinity" of the device with the corresponding IPv4 address. To state it in a more precise way, all the IPv6* devices whose addresses have a given v4* prefix must be "IPv6* routable" to/from the device that has the corresponding IPv4 address. The next 24 bits are used as a subnet identifier, allowing up to 2^{24} subnets within a single v4* prefix. The notation proposed to represent InRA /64 prefixes is 40:a.b.c.d:ABCDEF::/64, where a,b,c,d are decimal representations of one octet (a.b.c.d is therefore the conventional representation of an IPv4 address) and ABCDEF is an hexadecimal representation of 24 bits.

The relationship among the IPv6* and IPv4 address spaces makes possible the integration of their forwarding tables. The aim is to have a single consistent routing topology to avoid the problems resulting from the operation and management of two unrelated routing systems. This approach allows core Internet routers to remain unchanged while the edges migrate to the new routing model. However, new devices can manage prefixes more specific than 32 bits to obtain routes to the InRA hosts. As a result, the routing table of InRA-aware devices may contain three types of addresses: IPv4, IPv6 and IPv6*. Routes to IPv6 destinations would be available in IPv6 or InRA-aware IPv6* routers. Similarly routes to IPv4 destinations would be available in IPv4 and IPv6* routers.

Regarding to packet forwarding, the main requirement is that InRA packets should be forwarded not only by InRA routers, but also by IPv4-only devices. To allow this, the InRA packet contains the headers of both protocols by encapsulating an IPv6 header into an IPv4 one, as depicted in Fig. 2. IPv4 routers will use the IPv4 destination address and the IPv4 forwarding information to determine the next hop for the packet. On the other hand, IPv6* routers will use the IPv6* address contained in the inner header to perform forwarding. However, in order to handle properly IPv6 packets that have been tunnelled into IPv4 ones, a marking bit is needed. This marking bit should not be modified by IPv4-only devices, because otherwise confusion would arise on how to process the packet. Our proposal is to set to 1 the 48th bit of the header of InRA packets. This bit (informally called "evil bit") is not used by IPv4 and according to standards [10] and it must be forwarded unchanged by processing devices. InRA devices should set this bit to 1 when creating InRA packets. In this way, InRA devices can correctly identify and process an InRA packet, distinguishing it from conventional tunnelled IPv6 over IPv4 packets. IPv4-only devices can process InRA packets as conventional native IPv4 packets.

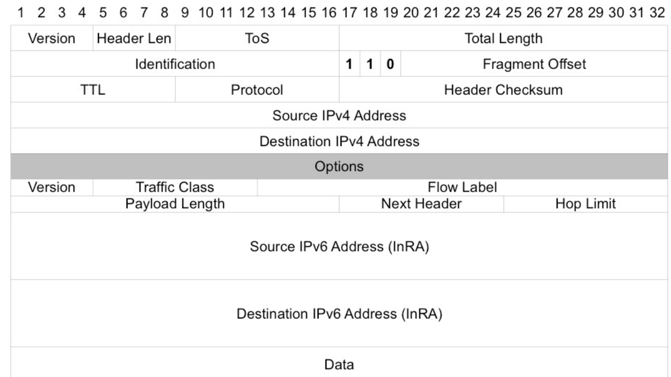

Fig. 2. Header of InRA packets.

We have carried out substantial experimental checks by sending IPv4 packets with this bit set to 1 across different paths of the Internet. In these experiments, the value of the TTL field of the packets ranged from 1 up to the number of hops to the destination. As a result, we received ICMP Time Exceeded messages from the routers on-path. The analysis of the received ICMP error message, in particular of the 64 bits containing the header of original datagram which raised the error, showed that all routers preserved the original value of the "evil bit".

New IPv6 deployment based on InRA should start from the edges: new networks should be assigned a single public IPv4 address, which would be used to generate the corresponding IPv6* address space. Internal routing would be based in /64 IPv6* prefixes. Note that dual-stack, the current recommended mechanism for IPv6 deployment, requires at least one public IPv4 address per network, so this requirement does not seems overwhelming. The connection of IPv6* domains across legacy IPv4 networks would rely on IPv4 routing. However, the InRA architecture supports the coexistence of different types of devices in a flexible way, as it is presented in the next section.

III. InRA Testbed

We now describe the experimental setup built to test InRA. For the test, we have developed a Click [11] implementation of InRA devices. We have used the sample topology and routing

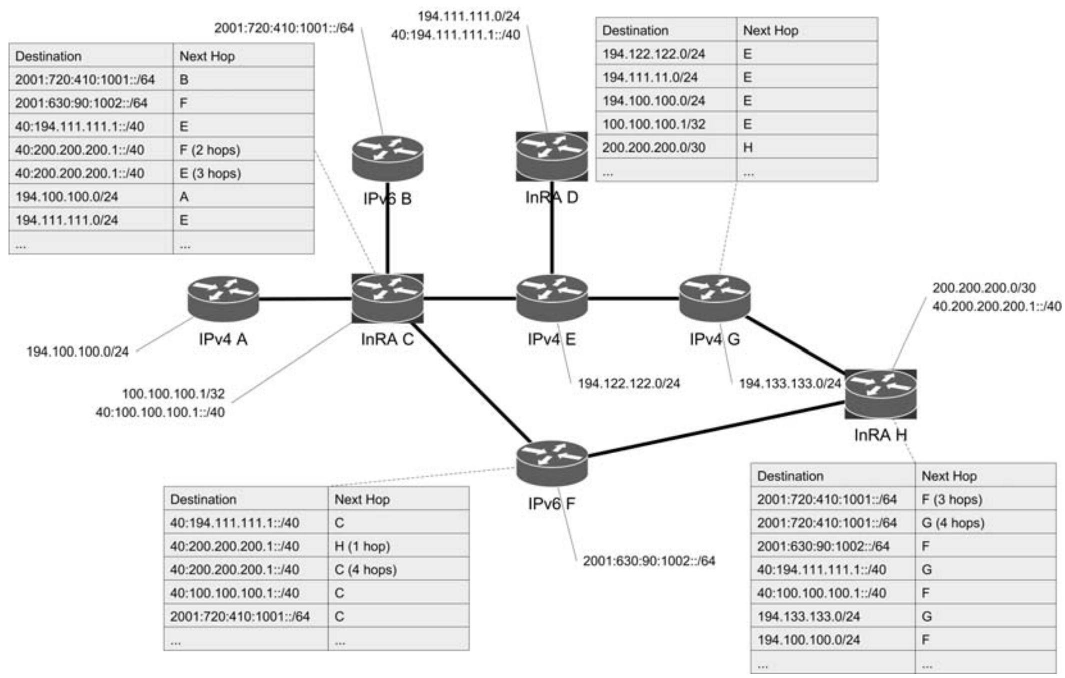

Fig. 3. Sample topology and routing tables with InRA.

tables presented in Fig. 3, comprised of IPv4, IPv6 and InRA-aware devices. It can be seen that InRA devices (such as C, D or H) have knowledge about the full topology including IPv4 and IPv6 routers.

Consider the case in which IPv6 packets are sent from router B to router H. Packets sent from router B have standard IPv6 syntax with the IPv6* destination address and are sent to InRA aware router C. By examining the destination address, router C can distinguish that it deals with IPv6* destination. Lets assume that router C prefers sending traffic to H across router F. In this case, router C forwards unchanged IPv6 packet through router F to router H. However, if this route becomes unavailable, the packet can be sent to router H through E and G, which are IPv4 routers. For this purpose, router C creates an IPv6* header which contains IPv4 and IPv6* destination address of H. This packet can easily be processed by IPv4 devices E and G. Once router H gets such a packet, it forwards it to the final destination according to entries containing the IPv6* address.

IV. Conclusions

Integrated Addressing and Routing can be seen as an extension to the 6to4 concept, in which we call for IPv6 routers to have a single, integrated, routing table. This implies a single integrated routing topology that on our experience would significantly cut the operational costs of a carrier company exploiting a hybrid IPv6 and IPv4 network.

The integration of routing and addressing strongly simplifies network management in comparison to a pure IPv6 approach. It is not necessary to maintain two addressing and routing schemes and hence routing topology and routing policies are not duplicated. It saves not only resources but also time spent on configuration and management. Because of the use of a single consistent scheme instead of two separated ones, InRA makes troubleshooting and reacting for potential problems easier and faster.

Acknowledgment

The work of Alberto García-Martínez was partially supported by the Spanish Ministerio de Ciencia e Innovación project grant TIN2008-06739-C04-01 (T2C2).

References

[1] G. Huston, "IPv4 address report." Available at: http://www.potaroo.net/tools/ipv4/index.html.
[2] L. Colitti, "Global IPv6 statistics - measuring the current state of IPv6 for ordinary users," *RIPE 57 Meeting*, Dubai, Oct. 2008.
[3] E. Nordmark and R. Gilligan, "Basic transition mechanisms for IPv6 hosts and routers," RFC4213, Oct. 2005.
[4] C. Huitema, "Teredo: tunneling IPv6 over UDP through network address translators NATs," RFC4380, Feb. 2006.
[5] M. Bagnulo, P. Matthews, and I. van Beijnum, "Stateful NAT64: network address and protocol translation from IPv6 clients to IPv4 servers," Internet Draft (work-in-progress),, Jan. 2010.
[6] E. Nordmark, "Stateless IP/ICMP translation algorithm (SIIT)," RFC2765, Feb. 2000.
[7] X. Li, C. Bao, M. Chen, H. Zhang, and J. Wu, "The CERNET IVI translation design and deployment for the IPv4/IPv6 coexistence and transition," Internet Draft (work-in-progress), Jan. 2010.
[8] J. Bi, J. Wi, and X. Leng, "IPv4/IPv6 transition technologies and univer6 architecture," *IJCSNS International J. Computer Sci. and Network Security*, vol. 7, no. 1, pp. 232–243, 2007.
[9] B. Carpenter and K. Moore, "Connection of IPv6 domains via IPv4 clouds," RFC3056, Feb. 2001.
[10] J. Postel, "Internet Protocol, Darpa Internet Program, protocol specification," RFC791, Sep. 1981.
[11] E. Kohler, R. Morris, B. Chen, J. Jannotti, and F. Kaashoek, "The click modular router," *ACM Trans. Computer Syst.*, vol. 18, no. 3, pp. 263–297, Aug. 2000.