IPv4 will remain sufficient in North America for the next 30 years

Rick Ramstetter
rick.ramstetter@gmail.com

Keywords:
IPv4, IPv6, RIR, IANA, NAT, CIDR

Abstract:
Since at least the introduction of Classless InterDomain Routing (CIDR) in 1993 it has been widely recognized that IPv4's lifetime is finite [1]. The four billion (2^32) addresses available therein are rapidly being consumed as a result of demands placed by emerging economies and ubiquitous computing platforms. The IETF and other governing bodies have put forth IPv6 as a proposed solution, though its deployment has been slower than necessary for sustained growth. [2]

In recent years IPv4's longevity has been extended through the use of new technologies. CIDR allows for highly efficient address space utilization, while Network Address Translation (NAT) facilitates multiple end users sharing a single globally visible IPv4 address. In some instances, large swathes of assigned but unused IPv4 address space have been reclaimed by governing bodies for redistribution.

These extensions to IPv4's lifetime have led many to question the urgency of deploying IPv6. While recognizing that IPv4's lifetime is indeed finite, I argue that such questioning is warranted, and that IPv4 will be sufficient in Norht America for at least the next 30 years. This argument has two components, the first of which is the technologies extending the life of IPv4 have not yet realized their full potential. The second component exploits the idea that *sufficient* need not be *prudent* or *optimal,* and that service providers are likely to do only that required to maintain sufficiency. That is to say IPv4 will be strictly *sufficient* indefinitely.

This paper is divided into seven sections. The first section will give some historical context to IPv4 and the IPv6 transition. The second section will summarize some of the estimates made regarding exhaustion of available IPv4 address reserves. The third through fifth sections will explain various technologies responsible for IPv4's extended lifetime and, if applicable, argue that the technology has not yet been fully exploited. The sixth section will introduce compatibility mechanisms between IPv6 and IPv4 and explain how even these mechanisms allow for the long term sufficiency of IPv4. The final section concludes by summarizing key ideas. Citations will be either intext or via footnotes.

History of IPv4 and the IPv6 transition:

The problem of uniquely identifying every machine or network connected to the Internet is not a new one. IPv4 was proposed via a Request for Comments (RFC) in September 1981 [3]. At this time, the term "Internet" was seven years new [4]. The Internet was controlled by the U.S. Federal Government and used primarily by educational institutions[5]. As part of the pioneering protocol suite of the internet, IPv4's designers could not possibly have anticipated many of the issues faced by IPv4 today. In addition to its limited address space, IPv4 is also plagued by large routing tables, limited support for secure transmission (lack of native IPSec), and

[1] IETF, RFCs 1517, 1519
[2] Huston, Geoff. "Confronting IPv4 address exhaustion". 09/2008.
[3] IETF, RFC 719
[4] Vinton Cerf's usage of the term "Internet" in IETF RFC675 is the earliest locatable written usage.
[5] Abbate, Janet. *Inventing the Internet*. Cambridge: MIT Press, 1999.

limited support for guaranteed quality of service (lack of native QoS)[6].

IPv6 natively fixes each of IPv4's issues (above). IPv6 is further a robust protocol that has been in development since the mid 1990s. Despite these facts its deployment has been slow in many areas. IPv4's flaws have thus far been addressed via site specific means.[7] These means are from an ISP's point of view externalized costs, whereas the transition to IPv6 is a purely internal cost that may be difficult to pass onto consumers due to government regulations. For example, the addition of NAT to a client network to combat a lack of cheaply available public IPv4 space is of little concern to the ISP involved. On the flipside, upgrading equipment and software to work with IPv6 to allow for cheaply available address space represents a large investment for the ISP[8]. A similar example can be found in QoS. A lack of QoS support in IPv4 has led many clients to simply overprovision their link capacity (resulting in higher profits for ISPs) rather than pressure their ISP for IPv6 and/or IPsec support[9]. The motivation for adoption of IPv6 is thus slim at best.

To further complicate matters, a lack of address space is of little concern to North American countries which, in total, control 53% of the IPv4 address space[10]. This large percentage can be attributed to the U.S. based origins of the internet. Of the Regional Internet Registries (RIRs)[11] the North American registry (ARIN) controls the most unallocated addresses[12]. Before the existence of RIRs address space was allocated directly by the Internet Assigned Numbers Authority (IANA) to requesting parties. The majority of these parties were North American companies. Nearly 50% of all available address space was given out in this fashion[13]. For example, General Electric, Ford Motor Company, and Prudential Securities each control an entire Class A[14] block of addresses (approximately 16 million IPv4 addresses each)[15]. In addition to tipping the allotment of available IP addresses to North America, such large swathes of address space are often underutilized. In the case of General Electric, 499 of 500 pseudorandom addresses were unresponsive to ping and connect attempts[16]. This indicates that either no machines are using these addresses or the machines to which the addresses are assigned do not communicate with the public Internet[17].

IPv4 Address Depletion estimates

Estimates on the exhaustion of available IPv4 addresses vary wildly. Older estimates often placed the exhaustion date in the late 1990s[18]. More recent estimates have placed the date somewhere between February 2009 and the year 2025[19]. The available data for analysis includes the history of IPv4 address assignments by RIRs and the history / frequency of relevant policy changes.

---

[6] From author's experience as a Systems' Administrator

[7] Huston, Geoff. "Confronting IPv4 address exhaustion". 09/2008.

[8] Gallaher, Michael P. IPV6 Economic Impact Assessment. Prepared for NIST. 10/2005.

[9] LópezOrtiz, Alejandro. "Valiant Load Balancing, Capacity Provisioning and Resilient Backbone Design." 2007.

[10] IPLigence.com Internet World Map, 2007 edition.

[11] Regional Internet Registry: Entity to whom various Internet Assigned Numbers Authority (IANA) powers are delegated on a regional basis.

[12] Geoff Huston, "Numerology." 11/05.

[13] "IPv4 Global Unicast Address Assignments", IANA

[14] Refers to classed network addressing. A Class A address block is of the form 1.x.x.x, a Class B is of the form 1.1.x.x, and a Class C of the form 1.1.1.x. An x indicates a field of the IP address in control of the assignee and can have a value from 0 to 255.

[15] IANA, IPv4 Global Unicast Address Assignments, 20080527

[16] A Bash shell script was written to choose a pseudorandom IP address beginning with 3 (that is, 3.x.x.x). A ping and connection to ports 80 and 21 were attempted to this IP via the Nmap utility. The script was executed 500 times via a loop.

[17] Address assignees are not required to use their non private addresses for communication over the public internet, though such is prudent.

[18] IETF, RFCs 1517, 1519

[19] Curran, John. Hearing before US House of Representatives Committee on Government Reform, "The next Generation Internet and the Transition to IPv6." 06/29/2005.

Worth noting is that the phrase "IPv4 address space exhaustion" may refer to either of two distinct events. The first is the date upon which IANA runs out of reserve addresses. The second is the date on which the RIRs run out of available addresses. [20] By definition IANA allocates IP addresses to RIRs; as such IANA will exhaust its reserve address pool before the RIRs.

IPv4 address assignees are not oblivious to the address pool's coming exhaustion. It is widely anticipated that there will be a "rush" of IPv4 assignments as the address pool approaches exhaustion[21]. To date, few if any statistical predictions regarding the exhaustion of IPv4 addresses take such a "rush" into account. Likewise, such predictions fail to take into account a presumed increase in the use of various address conserving or reclaiming technologies. As many predictions regarding the exhaustion of IPv4 addresses have passed without issue, it has become clear that such predictions can do little more than show the question at hand is not one of *if* but of *when.*

Classless routing

Much information on this topic is easily available; as such its introduction here will be brief. Classless Inter Domain Routing (CIDR) was introduced in 1993 to enable more efficient utilization of the global address space and shrink the size of routing tables. Rather than assigning a corporation a Class A, B, or C address block, the corporation is given control of a finely tuned number of addresses[22]. That is, the corporation is given control of a specifically sized *subnet.* This number is specified via a *subnet mask* essentially a bitmask for an IP address. The subnet mask is indicated via an integer number after the IP address and separated by a slash. See figure 1 for an example.

CIDR allows more efficient address space utilization by allowing for companies to be allocated only as much space as is required. A Class B address block contains 65,336 addresses, whereas a Class C contains 256 addresses. Prior to CIDR if a corporation had 260 machines it would need a Class B address block, thereby wasting (65336 – 260) addresses. CIDR allows this corporation to be assigned two contiguous Class C addresses a /23 assignment, where the first 23 bits of the address blocks are shared.

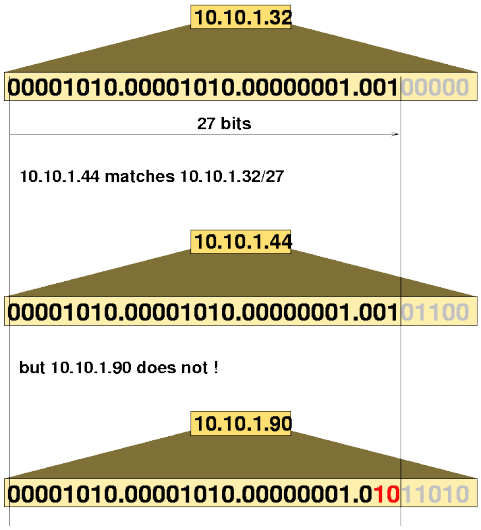

Image from Wikipedia, created by baccala@freesoft.org

Figure 1: classless address matching for 10.10.1.32/27 and 10.1.1.44/27. These IPs are identical for the first 27 bits, and are thus part of the same subnetwork.

CIDR has the additional benefit of reducing routing table size via route aggregation. The following example will illustrate. An ISP EarthNet has 8 customers, each of which has a Class C address block. Prior to CIDR, EarthNet would need to announce via BGP 8 Class C routes to the outside world. Assuming the 8 customers have contiguous IP address assignments, with CIDR EarthNet can announce one route for a /21 subnet. That is, EarthNet can announce that

[20] Since their creation RIRs have been responsible for assigning addresses to ISPs. Each of the RIRs hold a small reserve of IP addresses for assigning (typically about 32 million addresses). The global IP reserve is managed and allocated by IANA.

[21] Huston, Geoff. "IPv4 Address Lifetime Expectancy." 10/2006.

[22] IETF, RFCs 1518 and 1519

all addresses beginning with a certain 21 bits should be routed to it.

Network Address Translation (NAT) and private networks

Both IPv4 and IPv6 include the concept of nonglobally visible IP addresses (defined in RFC1918 and RFC1597). These addresses are not routed over the public internet instead being reserved for use with private networks. Common IPv4 private network addresses are of the form 10.x.x.x or 192.168.x.x. These and other private address ranges can be simultaneously used by disjoint entities. For example, many thousands of consumer grade routers each assign (via DHCP) addresses in the 192.168.x.x address block to connected machines[23]. By serving as a NAT gateway to the internet, any such router allows connected machines to share a single IP address.

Despite the existence of these private networks address ranges, approximately 42% of assigned, potentially publicly visible addresses are not visible to major Internet routers2. There are multiple explanations for this. Much of this 42% falls within the scope of addresses assigned by IANA before the creation of the RIRs and additionally before the creation of private address blocks. These IP address assignments are thus grandfathered in; they were assigned for private network use prior to the creation of private network blocks and today they remain in use on private networks. Of course, there is the additional possibility that such addresses are simply not in use.

Further, many of the RIRs (including ARIN) will assign public IP addresses for private network use to corporations that can demonstrate a need for private network address space beyond that available in typical private network address ranges [24]. That is, if a corporation has used up all addresses in the 10.x.x.x, 192.168.x.x, and other private address ranges, it can apply to its RIR for public IP addresses for use on its private network.

By definition IPv4 is a protocol for connecting disjoint networks. The model of connecting such networks via a gateway to a larger, global network (the Internet) allows NAT to be implemented with minimal architectural overhead. By implementing NAT at the network gateway an entire network can be made to use a single globally visible IPv4 address. A notable example of this is the Anaheim Union High School District of Orange County, CA. This district's 1,000+ personal computers share a single IPv4 address by utilizing NAT. 23

It is believed by the author that as IPv4 address reserves are depleted NAT usage will increase. Such has already happened in parts of the world without significant IPv4 address assignments. For example, in Europe it is common practice for ubiquitous computing platforms (ex: Cell Phones) to receive private network addresses[25]. The carrier then performs NAT on all of its customers; in the extreme all end users share a single globally visible IPv4 address. If United States' ISPs were to adopt a similar approach the potential address savings are innumerable.

A brief explanation of the ramifications of NAT on TCP and UDP is in order, though such is by definition a different network layer from either IPv4 or IPv6. A working knowledge of TCP and UDP is assumed.

NAT implementations generally allow for port forwarding of specific, externally available TCP/UDP ports to specific private network hosts. For example, public port 80 might forward to 10.1.1.2 (a corporate webserver), while public port 21 might forward to 10.1.1.3 (a corporate ftp server). Geoff Huston proposes that this concept can be extended to ISP/Carrierwide NAT (such as Vodafone). If the applications of each

[23] Author's experience as Systems Administrator

[24] ARIN Number Resource Policy Manual (IPv4), version 2008.4. 09/2008.

[25] Vodafone Mobile Connect Card Chooser Brochure, 8/5/2003.

client can limit themselves to 500 externally visible TCP/UDP ports a very reasonable bound then each globally visible IPv4 address can potentially sustain 130 clients. [26] If port triggering[27] were to be used, the number of clients sustainable by a single globally visible IPv4 address would be much larger. Most networked applications today allow users to select the TCP and UDP ports used. Such is of great benefit to ISPs: the cost of adding such functionality to applications is entirely external to them.

Given that the cost of adding NAT support to client applications is entirely external to ISPs, the cost of implementing NAT at an ISP level and divvying up TCP/UDP ports on a 500 per client basis is significantly less than deploying IPv6. [28] It has been argued that multilayered NAT has severe ramifications on application performance, and thus should be avoided (thereby ruling out ISPwide NAT). This argument fails to take into account that application robustness is purely external to ISPs.

Returning to IP, the ramifications of NAT on IPv4 are obvious: if IPv4 addresses can be reserved or reclaimed via the use of NAT, the usable lifetime of the protocol will be extended.

Opponents to widespread NAT usage argue that it hinders technological advancement by failing to provide a unique address for all end nodes. For example, 2002's RFC3439 states that NAT violates TCP/IPv4's "endto end principle." According to RFC3439's author, this principle states that "endtoend protocol design should not rely on the maintenance of state inside the network." Indeed, NAT usage requires that the NAT server keep track of inbound and outbound connections and destinations.

In addition to being written by someone not directly involved in the creation of the TCP/IPv4 protocol suite, this same RFC also states that "such state should be maintained only in the end points." In the model of disjoint networks connected to a global network it is not difficult to imagine the edge device connecting the two networks as an "end point." Such a device can be viewed as the end device ofa private network and beginning device of a global network. The NAT opponents' argument, then, is one of semantics. Did the protocol's designers intend for it to be used this way? Opponents would say "no." As already stated, though, IPv4's creators could not possibly have anticipated the problems IPv4 faces today. Thus, regardless of the creators' intent, the facts remains that widespread NAT usage promises significant benefits, and thathe potential market for NAT usage has not yet been saturated.

Address reclamation

Various authorities[29] encourage address block assignees to return unused address space. For example, BBN Technologies (now part of Level 3) recently returned two Class A address blocks. Unfortunately there is currently little, if any, motivation for corporations to return unused IP addresses. In the example of General Electric (assuming that it's 499 unresponsive IPs are in use but not publicly routed), it would take quite an effort for corporate Systems Administrators to assign new, private addresses to those systems an effort which lacks any financial incentive.

In some instances authorities have reclaimed IP address assignments known to be inefficiently utilized from their assignees. An extreme example was the 2006 reclamation of an entire Class A network,

[26] The number of available TCP/UDP ports, 65000, divided by 500 is 130.

[27] In a port triggered NAT environment, end user machines place requests via UPnP or SNMP to a NAT router that it forward specific ports to them. When the end user machine has finished using the ports, they are returned.

[28] NAT cost based on adding Cisco 7600 ISP grade, NAT capable routers at a cost of $20,000/each in addition to labor costs. IPv6 rollout cost based on replacing all equipment with IPv6 compatible replacements. NAT cost can potentially be lessened via the use of x86 hardware and open source software.

[29] IANA, ICANN, RIRs

specifically 14.0.0.0[30]. This address block was assigned to an older, infrequently used telephony protocol (X.25). Like the large corporations listed earlier, this address block contained approximately 16 million IP addresses.

There is no doubt that IP address reclamation is expensive. Leo Vegoda of IANA estimates it took 100 hours of labor to free up the 14.0.0.0 address block, in addition to the "5 minutes to 5 days" of effort put forth by address assignees. As IPv4 addressing currently exists there is no incentive to shoulder these costs. Many have proposed monetizing the IPv4 address system by introducing a market for buying and selling secondhand IPv4 addresses. This proposition wisely deserves attention, as it introduces financial incentive for conserving and returning IPv4 addresses while allowing for new Internet growth.

Many of the RIRs (ARIN, LACNIC, and RIPE to date) have adopted framework policies for the transfer of IPv4 address space between parties, though such policies will only become effective upon the exhaustion of all IPv4 address reserves. Preventing the monetization of IPv4 addresses until it is absolutely necessary is wise from an IPv6 perspective. Were IPv4 monetized today, address utilization efficiency would almost certainly skyrocket, thereby lessening demand for IPv6 addresses. That is to say, by introducing a system of monetizing IP address blocks IPv4 would become sufficient enough to further delay IPv6 deployment.

An additional prospect for monetization is monetizing BGP routing table entries. The operators of key Internet routers could, and rightfully should, charge to route multiple noncontiguous address spaces destined for a single end user via a single ISP. Such would result in address reallocation to accommodate greater routing efficiency. For example, suppose the ISPs EarthNet and PlutoNet exist. EarthCorp is a client of EarthNet with two Class B address ranges: 123.123.x.x and 012.012.x.x. These address blocks are not contiguous and thus cannot be announced to PlutoNet as a single route. Thus, EarthCorp requires two routing entries in PlutoNet's routing table. If PlutoNet were to assess a fee on EarthCorp for these multiple routes, EarthCorp would have incentive to acquire a contiguous address space (for example, sell rights to 012.012.x.x and purchase rights to 123.122.x.x).

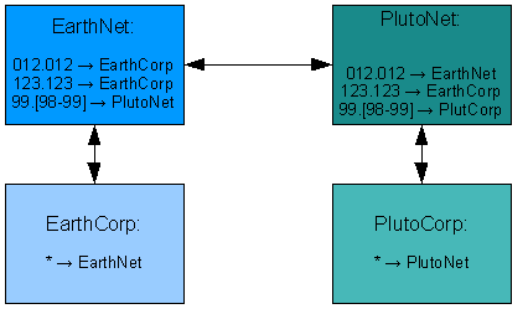

Figure 2: Both EarthCorp and PlutoCorp have two Class B address blocks. However, EarthCorp requires two routing table entries on PlutoNet, whereas PlutoCorp requires only one routing table entry on EarthNet. PlutoCorp's routing is more efficient as its address space is contiguous.

Returning to address space concerns, with 42% of available network addresses not routed on the public Internet, it cannot be claimed that reclamation is not possible or unsustainable. So long as reclaiming address space is marginally cheaper than migrating to IPv6 it will be a viable option. Given that IPv6 rollout involves significant new equipment (routers, for example) and training, significant IPv4 address space reclamation is entirely plausible.

IPv6 to IPv4 compatibility and transitional methods

IPv6 has been designed with significant backwards compatibility in mind. With an eye toward IPv4's prolonged sufficiency these compatibility mechanisms will be briefly introduced.

[30] Vegoda, Leo. "What was involved in reclaiming 14.0.0.0/8?" 09/2007.

Dual stack and IPv4 mapped addresses:

Dual stack deployments involve the use of both IPv4 and IPv6 on a given host. The choice between the two protocols is at the host's discretion, with communication preference generally given to IPv6 and falling back on IPv4 as necessary. For instance, on Windows XP SP2 machines operating in a dual stack environment, Internet Explorer 7.0 or later will initially attempt to make connections via IPv6 [31]. If that connection times out, an IPv4 attempt is made.

(As an aside, the author notes that such default behavior is poor at best. If an unknowing user enables an IPv6 interface under Windows XP, he might see his web performance decrease due to the previously described timeout. This user might then conclude that IPv6 is "slower" than IPv4 a logically sound but technically unfounded conclusion. Such misconceptions would serve only to further the long term demand for IPv4.)

On a dual stack host, the IPv4 address space exists inside the IPv6 address space as *IPv4 mapped addresses*. To rephrase, IPv4 mapped addresses represent IPv4 addresses as a subset of the IPv6 address space. These addresses allow application server software to operate in a dual stack environment with minimal overhead. For example, Example.com is a dual stack web server (it has both IPv4 and IPv6 addresses). The web server software (e.g. Apache) need only listen for incoming IPv6 connections; should an IPv4 connection to the server be made, the host operating system presents the incoming connection to Apache (or equivalent) as an IPv4 mapped address. (To clarify, IPv6's IPv4 mapped addresses are not seen "on the wire." Rather, the true IPv4 address is seen "on the wire.") Application servers (Apache), then, can easily communicate with both native IPv6 and native IPv4 clients. The point here is that dual stack servers (e.g. www.example.com) allow for the continued sufficiency of IPv4. Dual stack servers are particularly relevant to older operating systems like Windows 95 for which no IPv6 support exists.

IPv4 address	IPv4 mapped address under IPv6	IPv4 mapped address with dot decimal notation
192.0.2.128	::ffff:c000:280	::ffff:192.0.2.128
128.195.1.76	::ffff:80C3:CC	::ffff:128.195.1.76

Figure 3: IPv4 mapped address under IPv6

6to4, Teredo, and manual tunneling

Various tunneling protocols exist to allow indivudual IPv6 networks (networks supporting IPv6) to communicate over the IPv4only internet. The two basic types of tunneling available are manual and automatic. Manual tunneling is highly site specific and does not necessarily imply usage over IPv4; as such it will be excluded from explanation here.

Automatic tunneling protocols like 6to4 or Teredo are of particular interest. These protocols allow IPv6 tunnels over IPv4 to be created on demand. For example, 6to4 enabled hosts have an inherent ability to communicate directly with other 6to4 hosts[32]. The 6to4 host creates a virtual interface listening on IPv6. This interface accepts IPv6 connections from end user applications and encapsulates relevant incoming packets inside an IPv4 6in4 packet (where 6in4 is a minimalist protocol that simply pushes an IPv4 header onto the front of an IPv6 packet). These IPv4 packets are routed over the nonIPv6 public internet. Upon reaching the appropriate router, these packets are decapsulated and forwarded as appropriate in their true IPv6 form.

The Teredo protocol operates in much the same manner, with the primary differences being that UDP packets are used (rather than 6in4) to allow for NAT traversal, and that Teredo requires access to a specifically configured Teredo relay.

[31] Chown, Tim. "IPv6 transitioning and integration with IPv4

[32] Communication with "true" IPv6 hosts cannot take place without the help of a 6to4 relay.

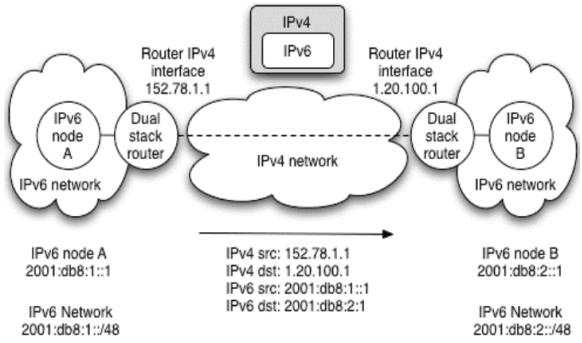

*Image from Tim Chown's "IPv6 transitioning and integration with IPv4", **tjc@ecs.soton.ac.uk***
Figure 4: IPv6 tunneling over an IPv4 network.

Ignoring implementation specific details, these tunneling protocols allow for the continued sufficiency of IPv4 by allowing for communication between IPv6 and IPv4 hosts. An IPv4 only host can utilize these or other IPv4/IPv6 tunneling protocols to create a virtual IPv6 interface. Should a server Example.com become IPv6 only sometime in the future, IPv4 clients can utilize a tunnel to enable communication with it.

Protocol Translation

Protocol Translation exists in many forms. The idea is to translate IPv6 address to, rather than tunnel them through, IPv4 addresses. Whereas tunneling can generally be done without tracking connection state in an external device, NAT Protocol Translation (NATPT) requires state information be recorded. NATPt operates in much the same fashion as IPv4 NAT, except translation between protocols also takes place. In the case of IPv4 only nodes trying to communicate with an IPv6 internet, a NATPT gateway device could statefully convert outbound IPv4 requests from nodes into IPv6 requests to the Internet, and similarly convert responses from IPv6 back to IPv4. The case of IPv6 only nodes communicating with an IPv4 only Internet can be handled similarly. By keeping track of connection state somewhere between the source and destination, NATPT allows for communication between IPv6 and IPv4 devices.

Much like NAT, the implementation of NAT PT represents an external cost to ISPs. This is an advantage of NATPT over Tunelling: at least some part of the cost of IPv6/IPv4 tunelling is internal to ISPs. For example, both 6to4 and Teredo require the use of external hosts to either forward Teredo traffic or provide for 6to4 decapsulation and forwarding. It is commonly thought that ISPs should provide such infrastructure.

Critics of NATPT would argue that it violates the end to end protocol and introduces unnecessary complexity. These arguments are very similar to the arguments against the use of NAT in general, and thus rebuttals to such arguments can be formed from the rebuttals to arguments against general NAT usage.

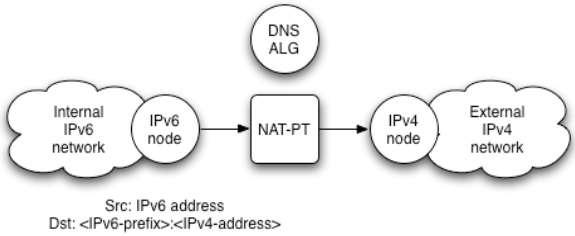

*Image from Tim Chown's "IPv6 transitioning and integration with IPv4", **tjc@ecs.soton.ac.uk***
Figure 5: Protocol translation via a network edge NAT PT device

An example of using NATPT to translate from IPv6 nodes to the IPv4 Internet follows. The NATPT router announces itself as the default route for all hosts on the internet. This router watches for outbound IPv6 DNS queries. When such a query is found, it is translated to an IPv4 DNS query and sent out to the Internet. When the IPv4 DNS query returns, the query is altered by the NATPT router to an IPv6 DNS response [33] and returned to the requesting node. By maintaining state information, the NATPT

[33] In altering the DNS response, the NATPT server changes the IPv6 address to one which cannot be routed over the public Internet. When the NATPT device sees connections to this address, it statefully knows how to make the appropriate translations.

device can translate client data requests via IPv6 to the appropriate IPv4 host.

Conclusion:

Each of the preceding technologies provides a means, or a part thereof, to allow for communication between IPv4 and IPv6 devices. By allowing for IPv4 encoded packets "on the wire," these technologies do not require the modification of the IP stacks on the countless routers between source and destination. That is, the preceding technologies allow the IPv4 internet to remain sufficient even when IPv6 only hosts (whether client or server) become popular or necessary.

The above technologies have allowed ISPs and hardware vendors to become stagnant. Utilizing these technologies, it is possible to deploy and manage a pure IPv6 network without any IPv6 transit to the Internet. Likewise, it is possible to deploy a multi thousand node IPv4 network which utilizes only a single public IPv4 address. These supposedly transitional methods have, in reality, crippled the motivating factors for an IPv6 transition and simultaneously ensured the continued sufficiency of IPv4.

To this day no manufacturer of home routing equipment has included IPv6 support in a North American home router. That new equipment is not being manufactured with IPv6 support speaks to the fact that address reclamation and IPv4 NAT will become more common.

Although the date upon which available IPv4 addresses are depleted cannot be known in advance, thanks to the technologies discussed in this paper it can be known that the exhaustion of IPv4's available address space does not correlate with the end of IPv4 usage. That is to say, IPv4 will remain sufficient even past the date upon which all IPv4 address reserves are exhausted.

When 802.11 technologies became popular, limitations of the TCP protocol in dealing with wireless networks were widely recognized. Though modifications to TCP were likely the most prudent means of solving the issues at hand, it was found to be an impossible task to upgrade entrenched TCP implementations. Thus, the 802.11 link layer must make provisions for handling TCP connections (such as link level transmission retry). In a similar fashion IPv4 is simply too well entrenched to change. The IPv4 protocol can be engineered around via the technologies explained in this paper, but it cannot be replaced.

Critics to the general position of this paper will argue that this adhoc mesh of technologies will result in a very "sloppy" Internet, and is not the optimal choice. I concede to this argument: it was never this paper's intention to show that IPv4's continued usage is optimal or prudent. The goal of this paper has been to argue that a technology as entrenched as IPv4 cannot be deprecated with any true meaning. Such an adhoc and sloppy Internet will indeed result in a need for extensive application layer support for NAT, NATPT, tunneling, etc. Unfortunately, these costs are purely external to transit providers.

That the costs of application support for NAT, NATPT, tunneling, etc, is external to ISPs is especially problematic given the high cost of migration to IPv6. For an ISP to transition to IPv6, routers and edge devices must be upgraded to IPv6 or replaced, either of which involves significant labor hours. Employees must be trained in the security and administration of IPv6. Compatibility issues with client networks must be resolved. From an economic standpoint, there is little motivation for an ISP to take on further expense (e.g. a transition of equipment to IPv6 compatibility) when failure to do so represents a wholly external cost.

As has been demonstrated ISPs will seek to externalize as many costs as is possible; they will seek only to provide as much functionality as is needed to remain

sufficient. Given that technologies exist to extend the lifetime of IPv4, and that technologies exist to enable communication between IPv4 and IPv6 hosts, IPv6 will not be viewed as necessary for sufficiency by entrenched ISPs in the near future, even if upstart ISPs in emerging economies use strictly IPv6. For better or worse, IPv4 will remain sufficient for the next 30+ years.

Further expansion upon this paper would entail an argument that a conversion to strictly IPv6 is the wrong goal, and that a more proper goal would be the seamless integration of IPv6 and IPv4 nodes.