Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 25, 2011

J. Arkko
Ericsson
M. Townsley
Cisco

October 22, 2010

IPv4 Run-Out and IPv4-IPv6 Co-Existence Scenarios
draft-arkko-townsley-coexistence-06

Abstract

When IPv6 was designed, it was expected that the transition from IPv4 to IPv6 would occur more smoothly and expeditiously than experience has revealed. The growth of the IPv4 Internet and predicted depletion of the free pool of IPv4 address blocks on a foreseeable horizon has highlighted an urgent need to revisit IPv6 deployment models. This document provides an overview of deployment scenarios with the goal of helping to understand what types of additional tools the industry needs to assist in IPv4 and IPv6 co-existence and transition.

This document was originally created as input to the Montreal co-existence interim meeting in October 2008, which led to the rechartering of the Behave and Softwire working groups to take on new IPv4 and IPv6 coexistence work. This document is published as a historical record of the thinking at the time, but hopefully also helps understand the rationale behind current IETF tools for co-existence and transition.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on April 25, 2011.

Table of Contents

1. Introduction

This document was originally created as input to the Montreal co-existence interim meeting in October 2008, which led to the rechartering of the Behave and Softwire working groups to take on new IPv4 and IPv6 coexistence work. This document is published as a historical record of the thinking at the time, but hopefully also helps understand the rationale behind current IETF tools for co-existence and transition.

When IPv6 was designed, it was expected that IPv6 would be enabled, in part or in whole, while continuing to run IPv4 side-by-side on the same network nodes and hosts. This method of transition is referred to as "Dual-Stack" [RFC4213] and has been the prevailing method driving the specifications and available tools for IPv6 to date.

Experience has shown that large-scale deployment of IPv6 takes time, effort, and significant investment. With IPv4 address pool depletion on the foreseeable horizon [Huston.IPv4], network operators and Internet Service Providers are being forced to consider network designs that no longer assume the same level of access to unique global IPv4 addresses. IPv6 alone does not alleviate this concern given the basic assumption that all hosts and nodes will be Dual-Stack until the eventual sunsetting of IPv4-only equipment. In short, the time-frames for the growth of the IPv4 Internet, the universal deployment of Dual-Stack IPv4 and IPv6, and the final transition to an IPv6-dominant Internet are not in alignment with what was originally expected.

While Dual-Stack remains the most well-understood approach to deploying IPv6 today, current realities dictate a re-assessment of the tools available for other deployment models that are likely to emerge. In particular, the implications of deploying multiple layers of IPv4 address translation need to be considered, as well as those associated with translation between IPv4 and IPv6 which led to the deprecation of [RFC2766] as detailed in [RFC4966]. This document outlines some of the scenarios where these address and protocol translation mechanisms could be useful, in addition to methods where carrying IPv4 over IPv6 may be used to assist in transition to IPv6 and co-existence with IPv4. We purposefully avoid a description of classic Dual-Stack methods, as well as IPv6 over IPv4 tunneling. Instead, this document focuses on scenarios which are driving tools we have historically not been developing standard solutions around.

It should be understood that the scenarios in this document represent new deployment models and are intended to complement, not replace existing ones. For instance, Dual-Stack continues to be the most recommended deployment model. Note that Dual-Stack is not limited to

situations where all hosts can acquire public IPv4 addresses. A common deployment scenario is running Dual-Stack on the IPv6 side with public addresses, and on the IPv4 side with just one public address and a traditional IPv4 NAT. Generally speaking, offering native connectivity with both IP versions is preferred over the use of translation or tunneling mechanisms when sufficient address space is available.

2. Scenarios

This section identifies five deployment scenarios which we believe have a significant level of near to medium term demand somewhere on the globe. We will discuss these in the following sections, while walking through a bit of the design space to get an understanding of the types of tools that could be developed to solve each. In particular, we want the reader to be consider what type of new equipment must be introduced in the network and where for each scenario, which nodes must be changed in some way, and which nodes must work together in an interoperable manner via a new or existing protocol.

The five scenarios are:

- Reaching the IPv4 Internet with less than one global IPv4 address per subscriber or subscriber household available (Section 2.1).
- Running a large network needing more addresses than those available in private RFC 1918 address space (Section 2.2).
- Running an IPv6-only network for operational simplicity as compared to Dual-Stack, while still needing access to the global IPv4 Internet for some, but not all, connectivity (Section 2.3).
- Reaching one or more privately addressed IPv4 only servers via IPv6 (Section 2.4).
- Accessing IPv6-only servers from IPv4 only clients (Section 2.5).

2.1. Reaching the IPv4 Internet

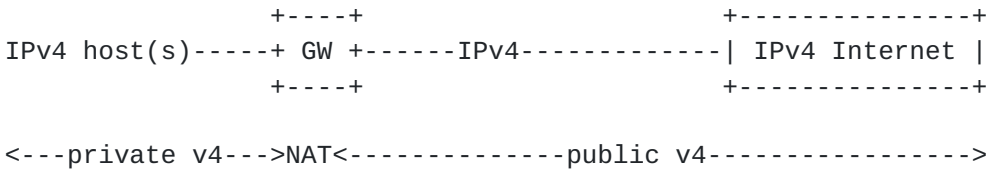


Figure 1: Accessing the IPv4 Internet today

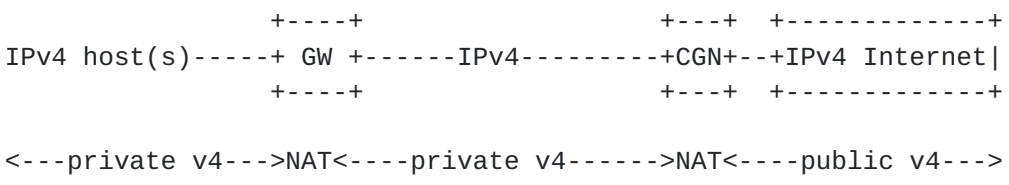
Figure 1 shows a typical model for accessing the IPv4 Internet today, with the gateway device implementing a Network Address and Port Translation (NAPT, or more simply referred to in this document as NAT). The NAT function serves a number of purposes, one of which is to allow more hosts behind the gateway (GW) than there are IPv4 addresses presented to the Internet. This multiplexing of IP addresses comes at great cost to the original end-to-end model of Internet, but nonetheless is the dominant method of access today, particularly to residential subscribers.

Taking the typical residential subscriber as an example, each subscriber line is allocated one global IPv4 address for it to use with as many devices as the NAT GW and local network can handle. As IPv4 address space becomes more constrained and without substantial movement to IPv6, it is expected that service providers will be pressured to assign a single global IPv4 address to multiple subscribers. Indeed, in some deployments this is already the case.

2.1.1. NAT444

When there is less than one address per subscriber at a given time, address multiplexing must be performed at a location where visibility to more than one subscriber can be realized. The most obvious place for this is within the service provider network itself, requiring the service provider to acquire and operate NAT equipment to allow sharing of addresses across multiple subscribers. For deployments where the GW is owned and operated by the customer, this becomes operational overhead for the Internet Service Provider (ISP) that it will no longer be able to rely on the customer and the seller of the GW device for.

This new address translation node has been termed a "Carrier Grade NAT", or CGN [I-D.nishitani-cgn]. The CGN's insertion into the ISP network is shown in Figure 2.



This solution approach is known as "NAT444" or "Double-NAT" and is discussed further in [I-D.wing-nat-pt-replacement-comparison].

It is important to note that while multiple levels of multiplexing of IPv4 addresses is occurring here, there is no aggregation of NAT state between the GW and CGN. Every flow that is originated in the subscriber home is represented as duplicate state in the GW and CGN. For example, if there are 4 PCs in a subscriber home, each with 25 open TCP sessions, both the GW and CGN must track 100 sessions each for that subscriber line.

NAT444 has the enticing property that it seems, at first glance, that the CGN can be deployed without any change to the GW device or other node in the network. While it is true that a GW which can accept a lease for a global IPv4 address would very likely accept a translated IPv4 address as well, the CGN is neither transparent to the GW or the subscriber. In short, it is a very different service model to offer a translated IPv4 address vs. a global IPv4 address to a customer. While many things may continue to work in both environments, some end-host applications may break, and GW port-mapping functionality will likely cease to work reliably. Further, if addresses between the subscriber network and service provider network overlap, ambiguous routes in the GW could lead to misdirected or black-holed traffic [I-D.shirasaki-isp-shared-addr]. Resolving this overlap through allocation of new private address space is difficult, as many existing devices rely on knowing what address ranges represent private addresses [I-D.azinger-additional-private-ipv4-space-issues].

Network operations which had previously been tied to a single IPv4 address for a subscriber would need to be considered when deploying NAT444 as well. These may include troubleshooting and OAM, accounting, logs (including legal intercept), QoS functions, anti-spoofing and security, backoffice systems, etc. Ironically, some of these considerations overlap with the kinds of considerations one needs to perform when deploying IPv6.

Consequences aside, NAT444 service is already being deployed in some networks for residential broadband service. It is safe to assume

that this trend will likely continue in the face of tightening IPv4 address availability. The operational considerations of NAT444 need to be well documented.

NAT444 assumes that the global IPv4 address offered to a residential subscriber today will simply be replaced with a single translated address. In order to try and circumvent performing NAT twice, and since the address being offered is no longer a global address, a service provider could begin offering a subnet of translated IPv4 addresses in hopes that the subscriber would route IPv4 in the GW rather than NAT. The same would be true if the GW was known to be an IP-unaware bridge. This makes assumptions on whether the ISP can enforce policies, or even identify specific capabilities, of the GW. Once we start opening the door to making changes at the GW, we have increased the potential design space considerably. The next section covers the same problem scenario of reaching the IPv4 Internet in the face of IPv4 address depletion, but with the added wrinkle that the GW can be updated or replaced along with the deployment of a CGN (or CGN-like) node.

2.1.2. Distributed NAT

Increasingly, service providers offering "triple-play" services own and manage a highly-functional GW in the subscriber home. These managed GWs generally have rather tight integration with the service provider network and applications. In these types of deployments, we can begin to consider what other possibilities exist besides NAT444 by assuming cooperative functionality between the CGN and GW.

If the connection between the GW and CGN is a point-to-point link (a common configuration between the GW and the "IP-Edge" in a number of access architectures), NAT-like functionality may be "split" between the GW and CGN rather than performing NAT444 as described in the previous section.

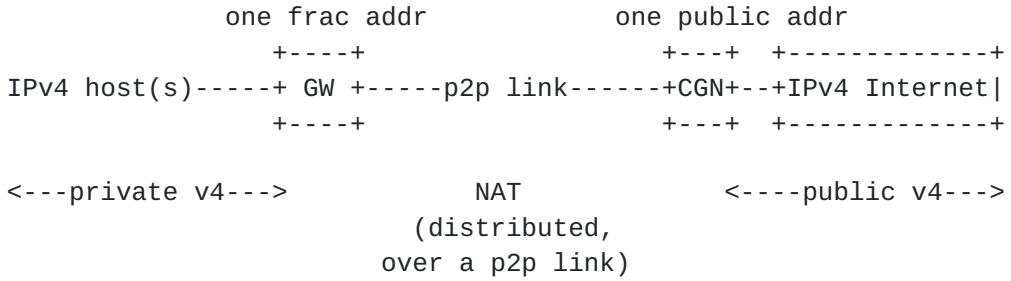


Figure 3: Distributed-NAT service

In this approach, multiple GWs share a common public IPv4 address, but with separate, non-overlapping, port ranges. Each such address/ port range pair is defined as a "fractional address". Each home gateway can use the address as if it were its own public address, except that only a limited port range is available to the gateway. The CGN is aware of the port ranges, which may be assigned in different ways, for instance during DHCP lease acquisition or dynamically when ports are needed [I-D.despres-v6ops-apbp]. The CGN directs traffic to the fractional address towards that subscriber's GW device. This method has the advantage that the more complicated aspects of the NAT function (Application Layer Gateways (ALGs), port-mapping, etc.) remain in the GW, augmented only by the restricted port-range allocated to the fractional address for that GW. The CGN is then free to operate in a fairly stateless manner, forwarding based on IP address and port ranges and not tracking any individual flows from within the subscriber network. There are obvious scaling benefits to this approach within the CGN node, with the tradeoff of complexity in terms of the number of nodes and protocols that must work together in an interoperable manner. Further, the GW is still receiving a global IPv4 address, albeit only a "portion" of one in terms of available port usage. There are still outstanding questions in terms of how to handle protocols that run directly over IP and cannot use the divided port number ranges, and handling of fragmented packets, but the benefit is that we are no longer burdened by two layers of NAT as in NAT444.

Not all access architectures provide a natural point to point link between the GW and CGN to tie into. Further, the CGN may not be incorporated into the IP Edge device in networks that do have point-to-point links. For these cases, we can build our own point-to-point link using a tunnel. A tunnel is essentially a point to point link that we create when needed [I-D.ietf-intarea-tunnels]. This is illustrated in Figure 4.

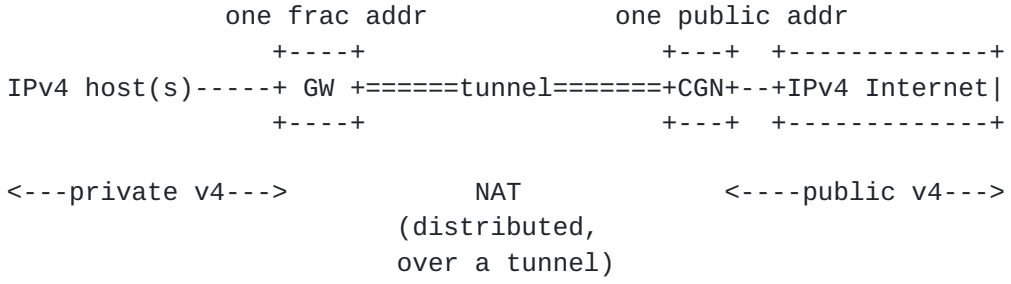


Figure 4: Point-to-point link created through a tunnel

Figure 4 is essentially the same as Figure 3, except the data link is created with a tunnel. The tunnel could be created in any number of ways depending on the underlying network.

At this point, we have used a tunnel or point-to-point link with coordinated operation between the GW and CGN in order to keep most of the NAT functionality in the GW.

Given the assumption of a point-to-point link between GW and CGN, the CGN could perform the NAT function, allowing private, overlapping, space to all subscribers. For example, each subscriber GW may be assigned the same 10.0.0.0/8 address space (or all RFC 1918 [RFC1918] space for that matter). The GW then becomes a simple "tunneling router" and the CGN takes on the full NAT role. One can think of this design as effectively a layer-3 VPN, but with Virtual-NAT tables rather than Virtual-Routing tables.

2.1.3. Recommendation

This section dealt strictly with the problem of reaching the IPv4 Internet with limited public address space for each device in a network. We explored combining NAT functions and tunnels between the GW and CGN to obtain similar results with different design tradeoffs. The methods presented can be summarized as:

a. Double-NAT (NAT444)

b. Single-NAT at CGN with a subnet and routing at the GW

c. Tunnel/link + Fractional IP (NAT at GW, port-routing at CGN)

d. Tunnel/link + Single NAT with overlapping RFC 1918 ("Virtual NAT" tables and routing at the GW)

In all of the above, the GW could be logically moved into a single host, potentially eliminating one level of NAT by that action alone. As long as the hosts themselves need only a single IPv4 address, methods b and d obviously are of little interest. This leaves methods a and c as the more interesting methods in cases where there is no analogous GW device (such as a campus network).

This document recommends the development of new guidelines and specifications to address the above methods. Cases where the home gateway both can and cannot be modified should be addressed.

2.2. Running out of IPv4 Private Address Space

In addition to public address space depletion, very large privately addressed networks are reaching exhaustion of RFC 1918 space on local networks as well. Very large service provider networks are prime candidates for this. Private address space is used locally in ISPs for a variety of things, including:

- control and management of service provider devices in subscriber premises (cable modems, set-top boxes, and so on) and
- addressing the subscriber's NAT devices in a double NAT arrangement, and
- "walled garden" data, voice, or video services.

Some providers deal with this problem by dividing their network into parts, each on its own copy of the private space. However, this limits the way services can be deployed and what management systems can reach what devices. It is also operationally complicated in the sense that the network operators have to understand which private scope they are in.

Tunnels were used in the previous section to facilitate distribution of a single global IPv4 address across multiple endpoints without using NAT, or to allow overlapping address space to GWs or hosts connected to a CGN. The kind of tunnel or link was not specified. If the tunnel used carries IPv4 over IPv6, the portion of the IPv6 network traversed naturally need not be IPv4 capable, and need not utilize IPv4 addresses, private or public, for the tunnel traffic to traverse the network. This is shown in Figure 5.

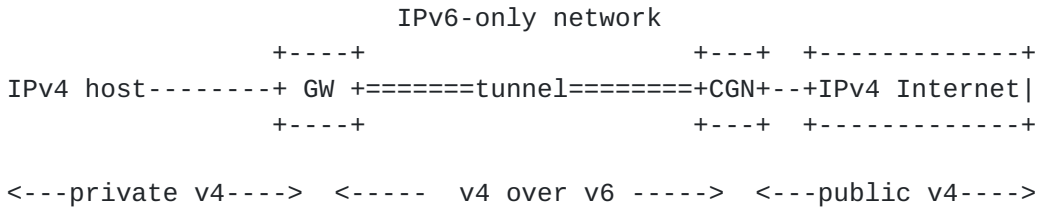


Figure 5: Running IPv4 over IPv6-only network

Each of the four approaches (a, b, c and d) from the Section 2.1 scenario could be applied here, and for brevity each iteration is not specified in full here. The models are essentially the same, just

that the tunnel is over an IPv6 network and carries IPv4 traffic. Note that while there are numerous solutions for carrying IPv6 over IPv4, this reverse mode is somewhat of an exception (one notable exception being the Softwire working group, as seen in [RFC4925]).

Once we have IPv6 to the GW (or host, if we consider the GW embedded in the host), enabling IPv6 and IPv4 over the IPv6 tunnel allows for Dual-Stack operation at the host or network behind the GW device. This is depicted in Figure 6:

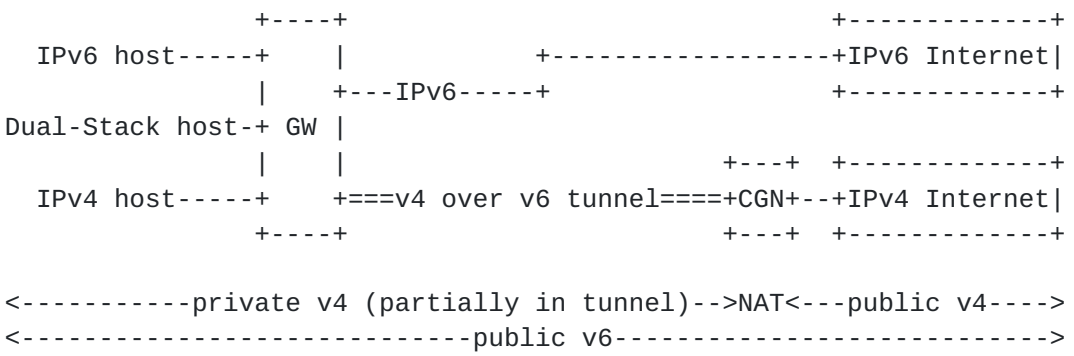


Figure 6: "Dual-Stack Lite" operation over an IPv6-only network

In [I-D.ietf-softwire-dual-stack-lite] this is referred to as "Dual-Stack Lite" bowing to the fact that it is Dual-Stack at the gateway, but not at the network. As introduced in Section 2.1, if the CGN here is a full functioning NAT, hosts behind a Dual-Stack Lite gateway can support IPv4-only and IPv6-enabled applications across an IPv6-only network without provisioning a unique IPv4 addresses to each gateway. In fact, every gateway may have the same address.

While the high-level problem space in this scenario is to alleviate local usage of IPv4 addresses within a service provider network, the solution direction identified with IPv6 has interesting operational properties that should be pointed out. By tunneling IPv4 over IPv6 across the service provider network, the separate problems of transition the service provider network to IPv6, deploying IPv6 to subscribers, and continuing to provide IPv4 service can all be decoupled. The service provider could deploy IPv6 internally, turn off IPv4 internally, and still carry IPv4 traffic across the IPv6 core for end users. In the extreme case, all of that IPv4 traffic need not be provisioned with different IPv4 addresses for each endpoint as there is not IPv4 routing or forwarding within the network. Thus, there are no issues with IPv4 renumbering, address space allocation, etc. within the network itself.

It is recommended that the IETF develop tools to address this scenario for both a host and GW. It is assumed that both endpoints of the tunnel can be modified to support these new tools.

2.3. Enterprise IPv6 Only Networks

This scenario is about allowing an IPv6-only host or a host which has no interfaces connected to an IPv4 network, to reach servers on the IPv4 internet. This is an important scenario for what we sometimes call "greenfield" deployments. One example is an enterprise network that wishes to operate only IPv6 for operational simplicity, but still wishes to reach the content in the IPv4 Internet. For instance, a new office building may be provisioned with IPv6-only. This is shown in Figure 7.

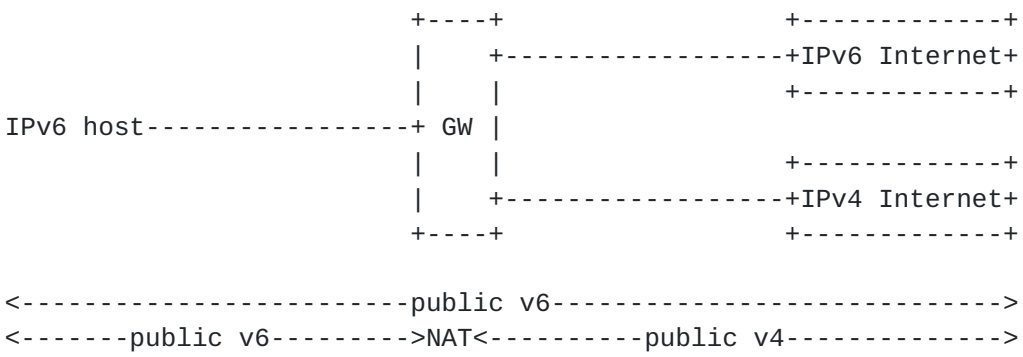


Figure 7: Enterprise IPv6-only network

Other cases that have been mentioned include "greenfield" wireless service provider networks and sensor networks. This bears a striking resemblance to Section 2.2 as well, if one considers the service provider network to simply be a very special kind of enterprise network.

In the Section 2.2 scenario, we dipped into design space enough to illustrate that the service provider was able to implement an IPv6-only network to ease their addressing problems via tunneling. This came at the cost of touching two devices on the edges of this network; both the GW and the CGN have to support IPv6 and the tunneling mechanism over IPv6. The greenfield enterprise scenario is different from that one in the sense that there is only one place that the enterprise can easily modify: the border between its network and the IPv4 Internet. Obviously, the IPv4 Internet operates the way it already does. But in addition, the hosts in the enterprise network are commercially available devices, personal computers with

existing operating systems. While we consider in this scenario that all of the devices on the network are "modern" Dual-Stack capable devices, we do not want to have to rely upon kernel-level modifications to these OSes. This restriction drives us to a "one box" type of solution, where IPv6 can be translated into IPv4 to reach the public Internet. This is one situation where new or improved IETF specifications could have an effect to the user experience in these networks. In fairness, it should be noted that even a network-based solution will take time and effort to deploy. This is essentially, again, a tradeoff between one new piece of equipment in the network, or a cooperation between two.

One approach to deal with this environment is to provide an application level proxy at the edge of the network (GW). For instance, if the only application that needs to reach the IPv4 Internet is the web, then a HTTP/HTTPS proxy can easily convert traffic from IPv6 into IPv4 on the outside.

Another more generic approach is to employ an IPv6 to IPv4 translator device. This is discussed in [I-D.wing-nat-pt-replacement-comparison]. NAT64 is an one example of a translation scheme falling under this category [I-D.ietf-behave-v6v4-framework] [I-D.ietf-behave-dns64] [I-D.ietf-behave-v6v4-xlate] [I-D.ietf-behave-v6v4-xlate-stateful] [I-D.ietf-behave-address-format].

Translation will in most cases have some negative consequences for the end-to-end operation of Internet protocols. For instance, the issues with Network Address Translation - Protocol Translation (NAT-PT) [RFC2766] have been described in [RFC4966]. It is important to note that the choice of translation solution and the assumptions about the network where they are used impact the consequences. A translator for the general case has a number of issues that a translator for a more specific situation may not have at all.

It is recommended that the IETF develop tools to address this scenario. These tools need to allow existing IPv6 hosts to operate unchanged.

2.4. Reaching Private IPv4 Only Servers

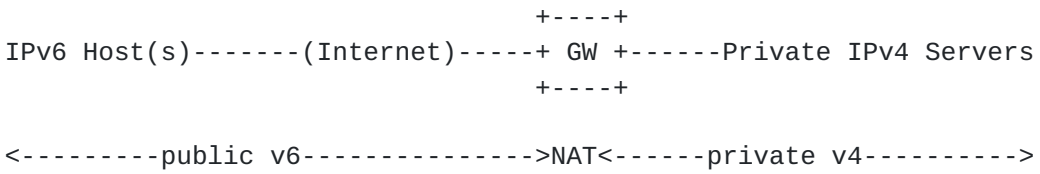
This section discusses the specific problem of IPv4-only capable server farms that no longer can be allocated a sufficient number of public addresses. It is expected that for individual servers, addresses are going to be available for a long time in a reasonably easy manner. However, a large server farm may require a large enough block of addresses that it is either not feasible to allocate one or it becomes economically desirable to use the addresses for other

purposes.

Another use case for this scenario involves a service provider that is capable of acquiring a sufficient number of IPv4 addresses, and has already done so. However, the service provider also simply wishes to start to offer an IPv6 service but without yet touching the server farm by upgrading it to IPv6.

One option available in such a situation is to move those servers and their clients to IPv6. However, moving to IPv6 is not just the cost of the IPv6 connectivity, but the cost of moving the application itself away to IPv6. So, in this case the server farm is IPv4 only, there is an increasing cost for IPv4 connectivity, and an expensive bill for moving server infrastructure to IPv6. What can be done?

If the clients are IPv4-only as well, the problem is a hard one, and dealt with in more depth in Section 2.5. However, there are important cases where large sets of clients are IPv6-capable. In these cases it is possible to place the server farm in private IPv4 space and arrange some of gateway service from IPv6 to IPv4 to reach the servers. This is shown in Figure 8.



One approach to implement this is to use NAT64 to translate IPv6 into private IPv4 addresses. The private IPv4 addresses are mapped into IPv6 addresses within a known prefix(es). The GW at the edge of the server farm is aware of the mapping, as is DNS. AAAA records for each server name is given an IPv6 address that corresponds to the mapped private IPv4 address. Thus, each privately addressed IPv4 server is given a public IPv6 presentation. No DNS application level gateway (DNS-ALG) is needed in this case, contrary to what NAT-PT required, for instance.

This is very similar to Section 2.3 where we typically think of a small site with IPv6 needing to reach the public IPv4 Internet. The difference here is that we assume not a small IPv6 site, but the whole of the IPv6 Internet needing to reach a small IPv4 site. This example was driven by the enterprise network with IPv4 servers, but

could be scaled down to the individual subscriber home level as well. Here, the same technique could be used to, say, access an IPv4 webcam in the home from the IPv6 internet. All that is needed is the ability to update AAAA records appropriately, an IPv6 client (which could use Teredo [RFC4380] or some other method to obtain IPv6 reachability), and the NAT64 mechanism. In this sense, this method looks much like a "NAT/FW bypass" function.

An argument could be made that since the host is likely Dual-Stack, existing port mapping services or NAT traversal techniques could be used to reach the private space instead of IPv6. This would have to be done anyway if the hosts are not all IPv6-capable or connected. However, in the case that they are, the alternative techniques force additional limitations on the use of port numbers. In the case of IPv6 to IPv4 translation, the full port space would be available for each server even in the private space.

It is recommended that the IETF develop tools to address this scenario. These tools need to allow existing IPv4 servers to operate unchanged.

2.5. Reaching IPv6 Only Servers

This scenario is predicted to become increasingly important as IPv4 global connectivity sufficient for supporting server-oriented content becomes significantly more difficult to obtain than global IPv6 connectivity. Historically, the expectation has been that for connectivity to IPv6-only devices, devices would either need to be IPv6 connected, or Dual-Stack with the ability to setup an IPv6 over IPv4 tunnel in order to access the IPv6 Internet. Many "modern" device stacks have this capability, and for them this scenario does not present a problem as long as a suitable gateway to terminate the tunnel and route the IPv6 packets is available. But, for the server operator, it may be a difficult proposition to leave all IPv4-only devices without reachability. Thus, if a solution for IPv4-only devices to reach IPv6-only servers were realizable, the benefits would be clear. Not only could servers move directly to IPv6 without trudging through a difficult Dual-Stack period, but they could do so without risk of losing connectivity with the IPv4-only Internet.

Unfortunately, realizing this goal is complicated by the fact that IPv4 to IPv6 is considered "hard" since of course IPv6 has a much larger address space than IPv4. Thus, representing 128 bits in 32 bits is not possible, barring the use of techniques similar to NAT64, which uses IPv6 addresses to represent IPv4 addresses as well.

The main questions about this scenario are about the timing and priority. While the expectation that this scenario may be of

importance one day is readily acceptable, at time of this writing there are little or no IPv6-only servers of importance beyond contrived cases that the authors are aware of. The difficulty of making a decision about this case is that, quite possibly, when there is sufficient pressure on IPv4 in order to see IPv6-only servers, the vast majority of hosts either have IPv6 connectivity, or the ability to tunnel IPv6 over IPv4 one way or another.

This discussion makes assumptions about what is a "server" as well. For the majority of applications seen on the IPv4 Internet to date, this distinction has been more or less clear. This is perhaps in no small part due to the overhead today in creating a truly end to end application in the face of the fragmented addressing and reachability brought on by the various NATs and firewalls employed today. This is beginning to shift, however, as we see more and more pressure to connect people to one another in an end-to-end fashion -- with peer-to-peer techniques, for instance -- rather than simply content server to client. Thus, if we consider an "IPv6-only server" as what we classically consider as an "IPv4 server" today, there may not be a lot of demand for this in the near future. However, with a more distributed model of the Internet in mind there may be more opportunities to employ IPv6-only "servers" that we would normally extrapolate based on past experience with applications.

It is recommended that IETF addresses this scenario, though perhaps with a slightly lower priority than the others. In any case, when new tools are developed to support this, it should be obvious that we cannot assume any support for updating legacy IPv4 hosts in order to reach the IPv6-only servers.

3. Security Considerations

Security aspects of the individual solutions are discussed in more depth elsewhere, for instance in [I-D.ietf-softwire-dual-stack-lite] [I-D.ietf-behave-v6v4-framework] [I-D.ietf-behave-dns64] [I-D.ietf-behave-v6v4-xlate] [I-D.ietf-behave-v6v4-xlate-stateful] [I-D.wing-nat-pt-replacement-comparison] [RFC4966]. This document highlights just three issues:

- Any type of translation may have an impact how certain protocols can pass through. For example, IPsec needs support for NAT traversal, and the proliferation of NATs implies an even higher reliance on these mechanisms. It may also require additional support for new types of translation.

- Some solutions have a need to modify results obtained from DNS. This may have an impact on DNS Security, as discussed in

[RFC4966]. Minimization or even elimination of such problems is essential, as discussed in [I-D.ietf-behave-dns64].

- o Tunneling solutions have their own security issues, for instance the need to secure tunnel endpoint discovery or to avoid opening up denial-of-service or reflection vulnerabilities [I-D.ietf-v6ops-tunnel-security-concerns].

4. IANA Considerations

This document has no actions for IANA.

5. Conclusions

The authors believe that the scenarios outlined in this document are among the top of the list of those that should to be addressed by the IETF community in short order. For each scenario, there are clearly different solution approaches with implementation, operations and deployment tradeoffs. Further, some approaches rely on existing or well-understood technology, while some require new protocols and changes to established network architecture. It is essential that these tradeoffs be considered, understood by the community at large, and in the end well-documented as part of the solution design.

After writing the initial version of this document, the Softwire working group was rechartered to address Section 2.2 scenario with a combination of existing tools (tunneling, IPv4 NATs) and some minor new ones (DHCP options) [I-D.ietf-softwire-dual-stack-lite]. Similarly, the Behave working group was rechartered to address scenarios from Section 2.3, Section 2.4, and Section 2.5. At the time this document is being published, proposals to address scenarios from Section 2.1 are still under consideration for new IETF work items.

This document set out to list scenarios that are important for the Internet community. While it introduces some design elements in order to understand and discuss tradeoffs, it does not list detailed requirements. In large part, the authors believe that exhaustive and detailed requirements would not be helpful at the expense of embarking on solutions given our current state of affairs. We do not expect any of the solutions to be perfect when measured from all vantage points. When looking for opportunities to deploy IPv6, reaching for perfection too far could become its own demise if we are not attentive to this. Our goal with this document is to support development of tools to help minimize the tangible problems that we are experiencing now, as well as those that we can best anticipate

down the road, in hopes of steering the Internet on its best course from here.

6. References

6.1. Normative References

[RFC1918] Rekhter, Y., Moskowitz, R., Karrenberg, D., Groot, G., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, February 1996.

[RFC4213] Nordmark, E. and R. Gilligan, "Basic Transition Mechanisms for IPv6 Hosts and Routers", RFC 4213, October 2005.

6.2. Informative References

[RFC2766] Tsirtsis, G. and P. Srisuresh, "Network Address Translation - Protocol Translation (NAT-PT)", RFC 2766, February 2000.

[RFC4380] Huitema, C., "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)", RFC 4380, February 2006.

[RFC4925] Li, X., Dawkins, S., Ward, D., and A. Durand, "Softwire Problem Statement", RFC 4925, July 2007.

[RFC4966] Aoun, C. and E. Davies, "Reasons to Move the Network Address Translator - Protocol Translator (NAT-PT) to Historic Status", RFC 4966, July 2007.

[I-D.wing-nat-pt-replacement-comparison] Wing, D., Ward, D., and A. Durand, "A Comparison of Proposals to Replace NAT-PT", Internet-Draft wing-nat-pt-replacement-comparison-00, September 2008.

[I-D.ietf-softwire-dual-stack-lite] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", draft-ietf-softwire-dual-stack-lite-06 (work in progress), August 2010.

[I-D.ietf-behave-v6v4-framework] Baker, F., Li, X., Bao, C., and K. Yin, "Framework for IPv4/IPv6 Translation", draft-ietf-behave-v6v4-framework-10 (work in progress), August 2010.

[I-D.ietf-behave-dns64]
Bagnulo, M., Sullivan, A., Matthews, P., and I. Beijnum, "DNS64: DNS extensions for Network Address Translation from IPv6 Clients to IPv4 Servers", draft-ietf-behave-dns64-11 (work in progress), October 2010.

[I-D.ietf-behave-v6v4-xlate]
Li, X., Bao, C., and F. Baker, "IP/ICMP Translation Algorithm", draft-ietf-behave-v6v4-xlate-23 (work in progress), September 2010.

[I-D.ietf-behave-v6v4-xlate-stateful]
Bagnulo, M., Matthews, P., and I. Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", draft-ietf-behave-v6v4-xlate-stateful-12 (work in progress), July 2010.

[I-D.ietf-behave-address-format]
Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", draft-ietf-behave-address-format-10 (work in progress), August 2010.

[I-D.ietf-intarea-tunnels]
Touch, J. and M. Townsley, "Tunnels in the Internet Architecture", draft-ietf-intarea-tunnels-00 (work in progress), March 2010.

[I-D.despres-v6ops-apbp]
Despres, R., "A Scalable IPv4-IPv6 Transition Architecture Need for an address-port-borrowing-protocol (APBP)", draft-despres-v6ops-apbp-01 (work in progress), July 2008.

[Huston.IPv4]
Huston, G., "The IPv4 Internet Report", available at http://ipv4.potaroo.net, August 2008.

[I-D.nishitani-cgn]
Nishitani, T., Miyakawa, S., Nakagawa, A., and H. Ashida, "Common Functions of Large Scale NAT (LSN)", draft-nishitani-cgn-02 (work in progress), June 2009.

[I-D.shirasaki-isp-shared-addr]
Shirasaki, Y., Miyakawa, S., Nakagawa, A., Yamaguchi, J., and H. Ashida, "ISP Shared Address", draft-shirasaki-isp-shared-addr-02 (work in progress),

March 2009.

[I-D.azinger-additional-private-ipv4-space-issues]
Azinger, M. and L. Vegoda, "Additional Private IPv4 Space Issues", draft-azinger-additional-private-ipv4-space-issues-04 (work in progress), April 2010.

[I-D.ietf-v6ops-tunnel-security-concerns]
Krishnan, S., Thaler, D., and J. Hoagland, "Security Concerns With IP Tunneling", draft-ietf-v6ops-tunnel-security-concerns-03 (work in progress), October 2010.

Appendix A. Acknowledgments

Discussions with a number of people including Dave Thaler, Thomas Narten, Marcelo Bagnulo, Fred Baker, Remi Depres, Lorenzo Colitti, Dan Wing, Brian Carpenter, and feedback during the Internet Area open meeting at IETF-72 were essential to the creation of the content in this document.

Authors' Addresses

Jari Arkko
Ericsson
Jorvas 02420
Finland

Email: jari.arkko@piuha.net

Mark Townsley
Cisco
Paris 75006
France

Email: townsley@cisco.com