Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 29, 2010

D. Thaler
Microsoft
L. Zhang
UCLA
G. Lebovitz
Juniper

October 26, 2009

# IAB Thoughts on IPv6 Network Address Translation
# draft-iab-ipv6-nat-02.txt

## Status of this Memo

## Copyright Notice

## Abstract

There has been much recent discussion on the topic of whether the IETF should develop standards for IPv6 Network Address Translators (NATs). This document articulates the architectural issues raised by IPv6 NATs, the pros and cons of having IPv6 NATs, and provides the IAB's thoughts on the current open issues and the solution space.

## Table of Contents

## 1. Introduction

In the past, the IAB has published a number of documents relating to Internet transparency and the end-to-end principle, and other IETF documents have also touched on these issues as well. These documents articulate the general principles on which the Internet architecture is based, as well as the core values that the Internet community seeks to protect going forward. Most recently, RFC 4924 [RFC4924] reaffirms these principles and provides a review of the various documents in this area.

Facing imminent IPv4 address space exhaustion, recently there have been increased efforts in IPv6 deployment. However, since late last year there have also been increased discussions about whether the IETF should standardize network address translation within IPv6. People who are against standardizing IPv6 NAT argue that there is no fundamental need for IPv6 NAT, and that as IPv6 continues to roll out, the Internet should converge towards reinstallation of the end-to-end reachability which has been a key factor in the Internet's success. On the other hand, people who are for IPv6 NAT believe that NAT vendors would provide IPv6 NAT implementations anyway as NAT can be a solution to a number of problems, and that the IETF should avoid repeating the same mistake as with IPv4 NAT, where the lack of protocol standards led to different IPv4 NAT implementations, making NAT traversal difficult.

An earlier effort, [RFC4864], provides a discussion of the real or perceived benefits of NAT and suggests alternatives for most of them, with the intent of showing that NAT is not required to get the desired benefits. However, it also identifies several gaps remaining to be filled.

This document provides the IAB's current thoughts on this debate. We believe that the issue at hand must be viewed from an overall architectural standpoint in order to fully assess the pros and cons of IPv6 NAT on the global Internet and its future development.

## 2. What is the Problem?

The discussions on the desire for IPv6 NAT can be summarized as follows. Network address translation is viewed as a solution to achieve a number of desired properties for individual networks: avoiding renumbering, facilitating multihoming, internal topology hiding, preventing host counting, and simple security. We discuss below each of these perceived benefits from NAT.

## 2.1. Avoiding Renumbering

As discussed in [RFC4864] Section 2.5, the ability to change service providers with minimal operational difficulty is an important requirement in many networks. However, renumbering is still quite painful today, as discussed in [I-D.carpenter-renum-needs-work]. Currently it requires reconfiguring devices that deal with IP addresses or prefixes, including DNS servers, DHCP servers, firewalls, IPsec policies, and potentially many other systems such as intrusion detection systems, inventory management systems, patch management systems, etc.

In practice today, renumbering does not seem to be a significant problem in consumer networks, such as home networks, where addresses or prefixes are typically obtained through DHCP, and are rarely manually configured in any component. However in managed networks, renumbering can be a serious problem.

We also note that many, if not most, large enterprise networks avoid the renumbering problem by using provider-independent (PI) IP address blocks. The use of PI addresses is inherent in today's Internet operations. However in smaller managed networks that cannot get provider-independent IP address blocks, renumbering remains a serious issue. Regional Internet Registries (RIRs) constantly receive requests for PI address blocks; one main reason that they hesitate in assigning PI address blocks to all users is the concern about the PI addresses' impact on the routing system scalability.

## 2.2. Site Multihoming

Another important requirement in many networks is site multihoming. A multihomed site essentially requires that its IP prefixes be present in the global routing table to achieve the desired reliability in its Internet connectivity as well as load balancing. In today's practice, multihomed sites with PI addresses announce their PI prefixes to the global routing system; multihomed sites with provider-allocated (PA) addresses also announce the PA prefix they obtained from one service provider to the global routing system through another service provider, effectively disabling provider-based prefix aggregation. This practice makes the global routing table scale linearly with the number of multihomed user networks.

This issue was identified in [RFC4864] Section 6.4. Unfortunately, no solution except NAT has been deployed today that can insulate the global routing system from the growing number of multihomed sites, where a multihomed site simply assigns multiple IPv4 addresses, one from each of its service providers, to its exit router which is an IPv4 NAT box. Using address translation to facilitate multihoming

support has one unique advantage: there is no impact on the routing system scalability, as the NAT box simply takes one address from each service provider, and the multihomed site does not inject its own routes into the system. Intuitively it also seems straightforward to roll the same solution into multihoming support in the IPv6 deployment.

However it is important to point out that a multihomed site announcing its own PI prefix(es) achieves important benefits that NAT-based multihoming support does not provide. Using PI addresses, end-to-end communications can be preserved in face of connectivity failures of individual service providers, as long as the site remains connected through at least one operational service provider. Announcing PI prefixes also gives a multihomed site the ability to perform traffic engineering and load balancing. While the users gain these benefits from PI-based multihoming, we also note that, in today's routing system, these gains are at the cost of the increased routing table size for all service providers.

### 2.3. Network Obfuscation

Most network administrators want to hide the details of the computing resources, information infrastructure, and communications networks within their borders. This desire is rooted in the basic security principle that an organization's assets are for its sole use and all information about those assets, their operation, and the methods and tactics of their use are proprietary secrets. Some organizations use their information and communication technologies as a competitive advantage in their industries. It is a generally held belief that measures must be taken to protect those secrets. The first layer of protection of those secrets is preventing access to the secrets or knowledge about the secrets whenever possible. It is understandable why network administrators would want to keep the details about the hosts on their network, as well as the network infrastructure itself, private. They believe that NAT helps achieve this goal.

#### 2.3.1. Hiding Hosts

As a specific measure of network obfuscation, network administrators wish to keep secret any and all information about the computer systems residing within their network boundaries. Such computer systems include workstations, laptops, servers, function-specific end-points (e.g., printers, scanners, IP telephones, point of sale machines, building door access-control devices), and such. They want to prevent an external entity from counting the number of hosts on the network. They also want to prevent host fingerprinting, i.e., gaining information about the constitution, contents, or function of a host. For example, they want to hide the role of a host, as

whether it is a user workstation, a finance server, a source code build server, or a printer. A second element of host fingerprinting prevention is to hide details that could aid an attacker in compromising the host. Such details might include the type of operating system, its version number, any patches it may or many not have, the make and model of the device hardware, any application software packages loaded, those version numbers and patches, and so on. With such information about hosts, an attacker can launch a more focused, targeted attack. Operators want to stop both host counting and host fingerprinting.

Where host counting is a concern, it is worth pointing out some of the challenges in preventing it. [Bellovin] showed how one can successfully count the number of hosts behind a certain type of simple NAT box. More complex NAT deployments, e.g., ones employing Network Address Port Translators (NAPTs) with a pool of public addresses that are randomly bound to internal hosts dynamically upon receipt of any new connection, and do so without persistency across connections from the same host are more successful in preventing host counting. However, the more complex the NAT deployment, the less likely that complex connection types like the Session Initiation Protocol (SIP) [RFC3261] and the Stream Control Transmission Protocol (SCTP) [RFC4960] will be able to successfully traverse the NAT. This observation follows the age-old axiom for networked computer systems: for every unit of security you gain, you give up a unit of convenience, and for every unit of convenience you hope to gain, you must give up a unit of security.

If fields such as fragment ID, TCP initial sequence number, or ephemeral port number are chosen in a predictable fashion (e.g., sequentially), then an attacker may correlate packets or connections coming from the same host.

To prevent counting hosts by counting addresses, one might be tempted to use a separate IP address for each transport-layer connection. Such an approach introduces other architectural problems, however. Within the host's subnet, various devices including switches, routers, and even the host's own hardware interface often have a limited amount of state available before causing communication using a large number of addresses to suffer significant performance problems. In addition, if an attacker can somehow determine an average number of connections per host, the attacker can still estimate the number of hosts based on the number of connections observed. Hence such an approach can adversely affect legitimate communication at all times, simply to raise the bar for an attacker.

Where host fingerprinting is concerned, even a complex NAT cannot prevent fingerprinting completely. The way that different hosts

respond to different requests and sequences of events will indicate consistently the type of a host that it is, its OS, version number, and sometimes applications installed, etc. Products exist that do this for network administrators as a service, as part of a vulnerability assessment.

These scanning tools initiate connections of various types across a range of possible IP addresses reachable through that network. They observe what returns, and then send follow-up messages accordingly until they "fingerprint" the host thoroughly. When run as part of a network assessment process, these tools are normally run from the inside of the network, behind the NAT. If such a tool is set outside a network boundary (as part of an external vulnerability assessment or penetration test) along the path of packets, and is passively observing and recording connection exchanges, over time it can fingerprint hosts only if it has a means of determining which externally viewed connections are originating from the same internal host. If the NATing is simple and static, and each host's internal address is always mapped to the same external address and vice versa, the tool has 100% success fingerprinting the host. With the internal hosts mapped to their external IP addresses and fingerprinted, the attacker can launch targeted attacks into those hosts, or reliably attempt to hijack those hosts' connections. If the NAT uses a single external IP, or a pool of dynamically assigned IP address for each host, but does so in a deterministic and predictable way, then the operation of fingerprinting is more complex, but quite achievable.

If the NAT uses dynamically assigned addresses, with short-term persistency, but no externally learnable determinism, then the problem gets harder for the attacker. The observer may be able to fingerprint a host during the lifetime of a particular IP address mapping, and across connections, but once that IP mapping is terminated, the observer doesn't immediately know which new mapping will be that same host. After much observation and correlation, the attacker could sometimes determine if an observed new connection in flight is from a familiar host. With that information, and a good set of man-in-the-middle attack tools, the attacker could attempt to compromise the host by hijacking a new connection of adequately long duration. If temporal persistency is not deployed on the NAT, then this tactic becomes almost impossible. As the difficulty and cost of the attack increases, the number of attackers attempting to employ it decreases. And certainly the attacker would not be able to initiate a connection toward a host for which the attacker does not know the current IP address binding. So the attacker is limited to hijacking observed connections thought to be from a familiar host, or to blindly initiating attacks on connections in flight. This is why network administrators appreciate complex NATs' ability to deter host counting and fingerprinting, but such deterrence comes at a cost of

host reachability.

### 2.4. Topology Hiding

It is perceived that a network operator may want to hide the details of the network topology, the size of the network, the identities of the internal routers, and the interconnection among the routers. This desire has been discussed in [RFC4864] Sections 4.4 and 6.2.

However the success of topology hiding is dependent upon the complexity, dynamism, and pervasiveness of bindings the NAT employs (all of which were described above). The more complex, the more the topology will be hidden, but the less likely that complex connection types will successfully traverse the NAT barrier. Thus the trade-off is reachability across applications.

Even if one can hide the actual addresses of internal hosts through address translation, this does not necessarily prove sufficient to hide internal topology. It may be possible to infer some aspects of topological information from passively observing packets. For example, based on packet timing, delay measurements, the Hop Limit field, or other fields in the packet header, one could infer the relative distance between multiple hosts. Once an observed session is believed to match a previously fingerprinted host, that host's distance from the NAT device may be learned, but not its exact location or particular internal subnet.

Host fingerprinting is required in order to do a thorough distance mapping. An attacker might then use message contents to lump certain types of devices into logical clusters, and take educated guesses at attacks. This is not, however, a thorough mapping. Some NATs change the TTL hop counts, much like an application-layer proxy would, while others don't; this is an administrative setting on more advanced NATs. The simpler and more static the NAT, the more possible this is. The more complex and dynamic and non-persistent the NAT bindings, the more difficult.

### 2.5. Summary Regarding NAT as a Tool for Network Obfuscation

The degree of obfuscation a NAT can achieve will be a function of its complexity as measured by:

- o The use of one-to-many NAPT mappings;
- o The randomness over time of the mappings from internal to external IP addresses, i.e., non-deterministic mappings from an outsider's perspective;
- o The lack of persistence of mappings, i.e., the shortness of mapping lifetimes and not using the same mapping repeatedly;

- o The use of re-writing in IP header fields such as TTL.

However, deployers be warned: as obfuscation increases, host reachability decreases. Mechanisms such as STUN [RFC5389] and Teredo [RFC4380] fail with the more complex NAT mechanisms.

### 2.6. Simple Security

It is commonly perceived that a NAT box provides one level of protection because external hosts cannot directly initiate communication with hosts behind a NAT. However one should not confuse NAT boxes with firewalls. As discussed in [RFC4864] Section 2.2, the act of translation does not provide security in itself, but rather the lack of pre-established or permanent filtering state. The stateful filtering function can provide the same level of protection without requiring a translation function. For further discussion, see [RFC4864] Section 4.2.

### 2.7. Discussion

At present, the primary benefits one may receive from deploying NAT appear to be avoiding renumbering and facilitating multihoming without impacting routing scalability.

Network obfuscation (host hiding, both counting and fingerprinting prevention, and topology hiding) may well be achieved with more complex NATs, but at the cost of losing some reachability and application success. Again, when it comes to security, this is often the case: to gain security one must give up some measure of convenience.

## 3. Architectural Considerations of IPv6 NAT

First it is important to distinguish between the effects of a NAT box vs. the effects of a firewall. A firewall is intended to prevent unwanted traffic [RFC4948] without impacting wanted traffic, whereas a NAT box also interferes with wanted traffic. In the remainder of this section, the term "reachability" is used with respect to wanted traffic.

The discussions on IPv6 NAT often refer to the wide deployment of IPv4 NAT, where people have both identified tangible benefits and gained operational experience. However the discussions so far seem mostly focused on the potential benefits that IPv6 NAT may, or may not, bring. Little attention has been paid to the bigger picture, as we elaborate below.

When considering the benefits that IPv6 NAT may bring to a site that deploys it, we must not overlook a bigger question: if one site deploys IPv6 NAT, what is the potential impact it brings to the rest of the Internet that does not do IPv6 NAT?  This important question does not seem to have been addressed, or addressed adequately.

We believe that the discussions on IPv6 NAT should be put in the context of the overall Internet architecture.  The foremost question is not how many benefits one may derive from using IPv6 NAT, but more fundamentally, whether a significant portion of parties on the Internet are willing to deploy IPv6 NAT, and hence whether we want to make IP address translation a permanent block in the Internet architecture.

One may argue that the answers to the above questions depend on whether we can find adequate solutions to the renumbering and multihoming problems.  It is worthwhile pointing out that IPv6 NAT is not the only solution to these two problems.  Renumbering can be avoided by allocating to users provider-independent addresses. Multihoming is already a pervasive practice today, not some new feature to be supported in the future, and NAT-based multihoming has serious limitations as discussed earlier.  The real issue is not multihoming per se, but the need for a scalable routing system.

If the answer to the above two questions is no, then non-IPv6-NAT parts of the world should *not* be affected by those sites that want to deploy IPv6 NAT.  More specifically, IPv6 users should be able to reach each other directly without having to worry about address translation boxes between the two ends.  IPv6 application developers in general should be able to program based on the assumption of end-to-end reachability (of wanted traffic), without having to address the issue of traversing NAT boxes.  For example, referrals and multi-party conversations are straightforward with end-to-end addressing, but vastly complicated in the presence of address translation. Similarly, network administrators should be able to run their networks without the added complexity of NATs, which can bring not only the cost of additional boxes, but also increased difficulties in network monitoring and problem debugging.

Given the diversity of the Internet user populations and the diversity in today's operational practice, it is conceivable that some parties may have a strong desire to deploy IPv6 NAT, and the Internet should accommodate different views that lead to different practices (i.e., some using IPv6 NAT, others not).

If we accept the view that some, but not all, parties want IPv6 NAT, then the real debate should not be on what benefits IPv6 NAT may bring to the parties who deploy it.  It is undeniable that network

address translation can bring certain benefits to its users. However, the real challenge we should address is how to design IPv6 NAT in such a way that it can hide its impact within some localized scope. If IPv6 NAT design can achieve this goal, then the Internet as a whole can strive for (re-installing) the end-to-end reachability model.

## 4. Solution Space

From an end-to-end perspective, the solution space for renumbering and multihoming can be broadly divided into three classes:

1. Endpoints get a stable, globally reachable address: In this class of solutions, end sites use provider-independent addressing and hence endpoints are unaffected by changing service providers. For this to be a complete solution, provider-independent addressing must be available to all managed networks (i.e., all networks that use manual configuration of addresses or prefixes in any type of system). However, in today's practice, assigning provider-independent addresses to all networks, including small ones, raises concerns with the scalability of the global routing system. This is an area of ongoing research and experimentation. In practice, network administrators have also been developing short-term approaches to resolve today's gap between the continued routing table growth and limitations in existing router capacity [NANOG].
2. Endpoints get a stable but non-globally-routable address on physical interfaces but a dynamic, globally routable address inside a tunnel: In this class of solutions, hosts use locally-scoped (and hence provider-independent) addresses for communication within the site. As a result, managed systems such as routers, DHCP servers, etc. all see stable addresses. Tunneling from the host to some infrastructure device is then used to provide the host with globally routable addresses which may change, but address changes are constrained to systems that operate over or beyond the tunnel, including DNS servers and applications. These systems, however, are the ones that often can already deal with changes today using mechanisms such as DNS dynamic update. However, if endpoints and the tunnel infrastructure devices are owned by different organizations, then solutions are harder to incrementally deploy due to the incentive and coordination issues involved.
3. Endpoints get a stable address which gets translated in the network: In this class of solutions, end sites use non-globally-routable addresses within the site, and translate them to globally routable addresses somewhere in the network. In general, this causes the loss of end-to-end transparency which is

the subject of [RFC4924] and the documents it surveys. If the translation is reversible, and the translation is indeed reversed by the time it reaches the other end of communication, then end-to-end transparency can be provided. However if the two translators involved are owned by different organizations, then solutions are harder to incrementally deploy due to the incentive and coordination issues involved.

Concerning routing scalability, although there is no immediate danger, routing scalability has been a long time concern in operational communities, and an effective and deployable solution must be found. We observe that the question at hand is not about whether some parties can run NAT, but rather, whether the Internet as a whole would be willing to rely on NAT to curtail the routing scalability problem, and whether we have investigated all the potential impacts of doing so to understand its cost on the overall architecture. If effective solutions can be deployed in time to allow assigning provider-independent IPv6 addresses to all user communities, the Internet can avoid the complexity and fragility and other unforeseen problems introduced by NAT.

## 4.1. Discussion

As [RFC4924] states:

> A network that does not filter or transform the data that it carries may be said to be "transparent" or "oblivious" to the content of packets. Networks that provide oblivious transport enable the deployment of new services without requiring changes to the core. It is this flexibility that is perhaps both the Internet's most essential characteristic as well as one of the most important contributors to its success.

We believe that providing end-to-end transparency, as defined above, is key to the success of the Internet. While some fields of traffic (e.g., Hop Limit) are defined to be mutable, transparency requires that fields not defined as such arrive un-transformed. Currently, the source and destination addresses are defined as immutable fields, and are used as such by many protocols and applications.

Each of the three classes of solution can be defined in a way that preserves end-to-end transparency. We strongly encourage the community to consider end-to-end transparency as a requirement when proposing any solution, whether it be based on tunneling or translation or some other technique. Solutions can then be compared based on other aspects such as scalability and ease of deployment.

## 5. Security Considerations

Section 2 discusses potential privacy concerns as part of the Host Counting and Topology Hiding problems.

## 6. IANA Considerations

[RFC Editor: please remove this section prior to publication.]

This document has no IANA Actions.

## 7. IAB Members at the time of this writing

Marcelo Bagnulo
Gonzalo Camarillo
Stuart Cheshire
Vijay Gill
Russ Housley
John Klensin
Olaf Kolkman
Gregory Lebovitz
Andrew Malis
Danny McPherson
David Oran
Jon Peterson
Dave Thaler

## 8. References

### 8.1. Normative References

### 8.2. Informative References

[Bellovin]
Bellovin, S., "A Technique for Counting NATted Hosts", Proc. Second Internet Measurement Workshop , November 2002, <http://www.cs.columbia.edu/~smb/papers/fnat.pdf>.

[I-D.carpenter-renum-needs-work]
Carpenter, B., Atkinson, R., and H. Flinck, "Renumbering still needs work", draft-carpenter-renum-needs-work-04 (work in progress), October 2009.

[NANOG] "Extending the Life of Layer 3 Switches in a 256k+ Route

World", NANOG 44 , October 2008, <http://www.nanog.org/meetings/nanog44/presentations/Monday/Roisman_lightning.pdf>.

[RFC3041] Narten, T. and R. Draves, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 3041, January 2001.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

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

[RFC4864] Van de Velde, G., Hain, T., Droms, R., Carpenter, B., and E. Klein, "Local Network Protection for IPv6", RFC 4864, May 2007.

[RFC4924] Aboba, B. and E. Davies, "Reflections on Internet Transparency", RFC 4924, July 2007.

[RFC4948] Andersson, L., Davies, E., and L. Zhang, "Report from the IAB workshop on Unwanted Traffic March 9-10, 2006", RFC 4948, August 2007.

[RFC4960] Stewart, R., "Stream Control Transmission Protocol", RFC 4960, September 2007.

[RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT (STUN)", RFC 5389, October 2008.

## Authors' Addresses

Dave Thaler
Microsoft Corporation
One Microsoft Way
Redmond, WA  98052
USA

Phone: +1 425 703 8835
Email: dthaler@microsoft.com

Lixia Zhang
UCLA Computer Science Department
3713 Boelter Hall
Los Angeles, CA  90095
USA

Phone: +1 310 825 2695
Email: lixia@cs.ucla.edu

Gregory Lebovitz
Juniper Networks, Inc.
1194 North Mathilda Ave.
Sunnyvale, CA  94089
USA

Email: gregory.ietf@gmail.com