BEHAVE WG
Internet-Draft
Intended status: Standards Track
Expires: May 5, 2009

M. Wasserman
Sandstorm Enterprises
F. Baker
Cisco Systems

November 2008

IPv6-to-IPv6 Network Address Translation (NAT66)
draft-mrw-behave-nat66-02.txt

Status of this Memo

Copyright Notice

Abstract

This document describes a stateless, transport-agnostic IPv6-to-IPv6 Network Address Translation (NAT66) function that provides the

address independence benefit associated with IPv4-to-IPv4 NAT (NAT44) while minimizing, but not completely eliminating, the problems associated with NAT44.

Table of Contents

1. Requirements Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

2. Introduction

This document describes a stateless, transport-agnostic IPv6-to-IPv6 Network Address Translation (NAT66) function that provides the address independence benefit associated with IPv4-to-IPv4 NAT (NAT44) while minimizing, but not completely eliminating, the problems associated with NAT44.

3. What is Address Independence?

As used in this document, IPv6 Address Independence consists of the following set of local network properties:

- The IPv6 addresses in use inside the local network (for nodes, ACLs, logs) do not need to be renumbered if the ISP changes a site's external address prefix.

- The IPv6 addresses in use inside the local network (for nodes, ACLs, logs) do not need to be renumbered when a site changes ISPs.

- It is not necessary for an administrator to convince an ISP to route his or her internal IPv6 addresses.

This address independence requirement has been a primary driver for IPv4 NAT deployment in medium to large-sized enterprise networks, including NAT deployments in enterprises that have plenty of IPv4 provider-independent address space (from IPv4 "swamp space").

The Local Network Protection document [RFC4864] discusses a related concept called "Address Autonomy" as a benefit of NAT44. RFC 4864 indicates that address autonomy can be achieved by the simultaneous use of global addresses on all nodes within a site that need external connectivity, and Unique Local Addresses (ULAs) [RFC4193] for all internal communication. However, this solution fails to meet the requirement for address independence, because if an ISP renumbering event occurs, all of the hosts, routers, DHCP servers, ACLs, firewalls and other internal systems that are configured with global addresses from the ISP will need to be renumbered before global connectivity is fully restored.

The use of IPv6 Provider Independent (PI) addresses has also been suggested as a means to fulfill the address independence requirement. However, this solution requires that an enterprise qualify to receive a PI assignment and persuade their ISP to install specific routes for the enterprise's PI addresses. There are a number of practical issues with this approach, especially if there is a desire to route to a number of geographically and topologically diverse set of sites, which can sometimes involve coordinating with several ISPs to route portions of a single PI prefix. These problems have caused numerous enterprises with plenty of IPv4 swamp space to choose to use IPv4 NAT for part, or substantially all, of their internal network instead of using their provider-independent address space.

4. NAT66 Applicability

NAT66 provides a simple and compelling solution to meet the Address Independence requirement in IPv6. The address independence benefit stems directly from the translation function of the network address translator. To avoid as many of the issues associated with NAT44 as possible, NAT66 is defined to include a two-way, checksum-neutral, algorithmic translation function, and nothing else.

NAT66 does not include a port mapping function, and the defined address mapping mechanism is checksum-neutral. This avoids the need for a NAT66 device to re-write transport layer headers, making it feasible to deploy new or improved transport layer protocols without upgrading NAT66 devices. Because NAT66 does not involve re-writing transport-layer headers, NAT66 will not interfere with encrypting the full IP payload in many cases.

The default NAT66 address mapping mechanism is purely algorithmic, so NAT66 devices do not need to maintain per-node or per-connection state, allowing deployment of more robust and adaptive networks than can be deployed using NAT44. Since the default NAT66 mapping can be performed in either direction, it does not interfere with inbound connection establishment, thus allowing internal nodes to participate in direct peer-to-peer applications.

Although NAT66 compares favorably to NAT44 in several ways, it does not eliminate all of the architectural problems associated with IPv4 NAT, as described in [RFC2993]. NAT66 involves modifying IP headers in transit, so it is not compatible with security mechanisms that involve end-to-end encryption of the IP header, or mechanisms, such as AH, that provide integrity protection for the IP header. NAT66 may interfere with the use of application protocols that transmit IP addresses in the application-specific portion of the IP packet. These applications currently require application layer gateways

(ALGs) to work correctly through NAT44 devices, and similar ALGs may be required for these applications to work through NAT66 devices. The use of separate internal and external address prefixes creates complexity for DNS deployment, due the desire for internal nodes to communicate with other internal nodes using internal addresses, while external nodes need to obtain external addresses to communicate with the same nodes. Typically, this results in the deployment of "split DNS", which mad add complexity to network configuration.

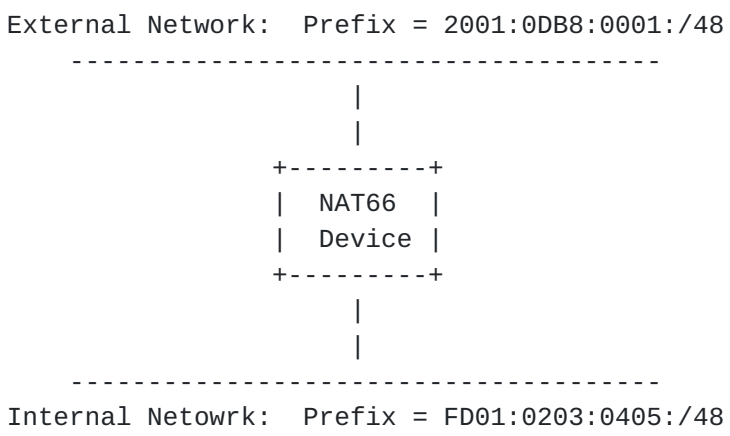
There are significant technical impactss associated with the deployment of any address translation mechanism, including NAT66, and we strongly encourage anyone who is considering the implementation or deployment of NAT66 to read RFC 4864 [RFC4864], and to carefully consider the alternatives described in that document, many of which will cause fewer problems than NAT66.

5. NAT66 Overview

NAT66 may be implemented in an IPv6 router to map one IPv6 address prefix to another IPv6 address prefix as each IPv6 packet transits the router. A router that implements a NAT66 function is referred to as a NAT66 device.

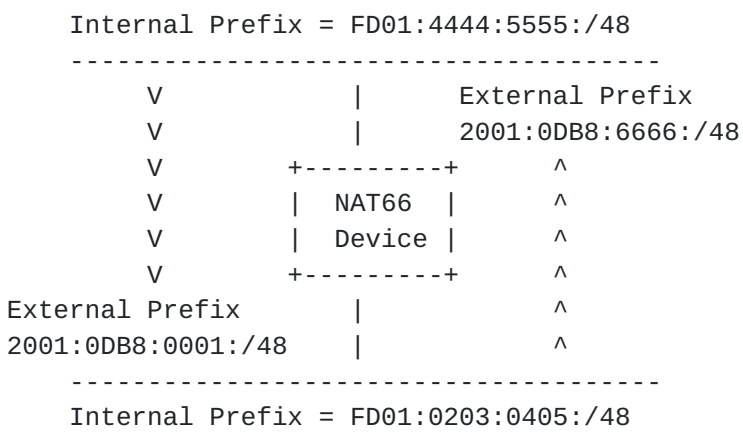
In its simplest form, a NAT66 device will be attached to two network links, one of which is an "internal" network link attached to a leaf network within a single administrative domain, and the other of which is an "external" network with connectivity to the global Internet. All of the hosts on the internal network will use addresses from a single, locally-routed prefix, and those addresses will be translated to/from addresses in a globally-routable prefix as IP packets transit the NAT66 device.

The following picture shows a NAT66 device attached to two networks. In this example, the internal network uses IPv6 Unique Local Addresses (ULAs) [RFC4193] to represent the internal IPv6 nodes, and the external network uses globally routable IPv6 addresses to represent the same nodes.

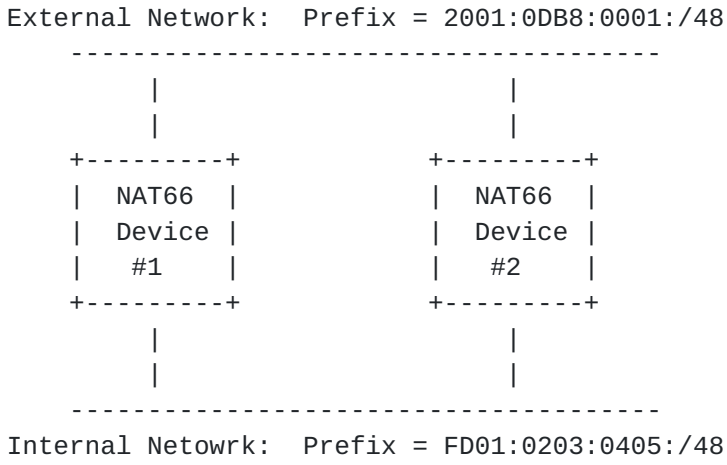


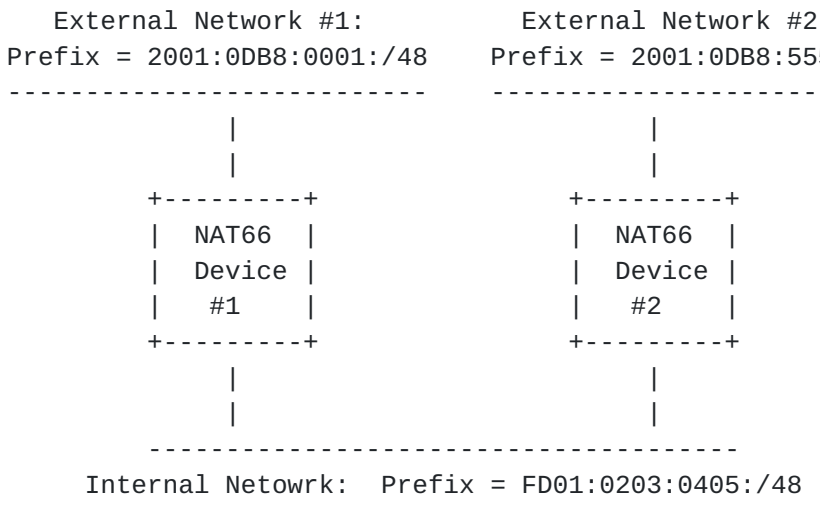
When a NAT66 device forwards packets in the "outbound" direction, from the internal network to the external network, NAT66 overwrites the IPv6 source address (in the IPv6 header) with a corresponding address from the external prefix. When packets are forwarded in the "inbound" direction, from the external network to the internal network, the IPv6 destination address is overwritten with a corresponding address in the internal prefix. Using the prefixes shown in the diagram above, as an IP packet passes through the NAT66 device in the outbound direction, the source address prefix (FD01:0203:0405:/48) will be overwritten with the external address prefix (2001:0DB8:0001:/48). In an inbound packet, the destination prefix (2001:0DB8:0001:/48) will be overwritten with the internal network prefix (FD01:0203:0405:/48). In both cases, it is the local IPv6 address that is overwritten; the remote IPv6 address remains unchanged. Nodes on the the mapped network are said to be "behind" the NAT66 device.

NAT66 can also be used between two private networks. In these cases, both networks may use ULA prefixes, with each subnet in one network mapped into a corresponding subnet in the other network, and vice versa. Or, each network may use ULA prefixes for internal addressing and global unicast addresses on the other network.



In some cases, more than one NAT66 device may be attached to a network. In this case, they two NAT66 devices may be configured with the same internal and external prefixes, or they may be configured with the same internal prefix and different external prefixes.





6. NAT66 Address Mapping

When NAT66 is used as described in this document, no per-node or per-flow state is maintained in the NAT66 device. Both inbound and outbound packets are translated algorithmically, using only information found in the IPv6 header. Due to this property, NAT66's two-way, algorithmic address mapping can support both outbound and inbound connection establishment without the need for state-priming or rendevous mechanisms. This is a significant improvement over NAT44 devices, but it also has significant security implications which are described in the Security Considerations section.

6.1. Checksum-Neutral Mapping

Note: There has been some discussion of whether it is necessary or even desireable to have NAT66 use a checksum-neutral mapping.

When a change is made to one of the IP header fields in the IPv6 pseudo-header checksum (such as the IP addresses), the checksum field in the transport layer header may become invalid. Fortunately, an incremental change in the area covered by the Internet standard checksum [RFC1071] will result in a well-defined change to the checksum value [RFC1624]. So, a checksum change caused by modifying one part of the area covered by the checksum can be eliminated by making a complementary change to a different 16-bit field covered by the same checksum.

The NAT66 address mapping mechanism described in this document is checksum-neutral, which means that it results in IP headers that will generate the same pseudo-header checksum when the checksum is

calculated using the standard Internet checksum [RFC1071]. Any changes that are made during translation of the IPv6 prefix are offset by changes to other parts of the IPv6 address. This results in the transport layers that use the Internet checksum (such as TCP and UDP) calculating the same IPv6 pseudo header checksum for both the internal and external forms of the same packet, which avoids the need for the NAT66 device to modify those transport layer headers.

To produce a checksum neutral transformation, the NAT66 device calculates the 16-bit one's complement sum of the internal and external IPv6 prefixes. The difference between the original and mapped prefix checksums is calculated using 16-bit one's complement arithmetic, and the difference is added to the 16-bit value in bits 49-64 of the mapped IPv6 address. If the resulting value is 0xFFFF, it is changed to 0x0000. Use of this algorithm results in an IPv6 header that will have the same pseudo-header checksum as the original header. Although the same mechanism is used to ensure that the NAT66 mapping is checksum-neutral, the corresponding change may be made in different locations depending upon the prefix length.

6.2. Address Mapping Example

The NAT66 two-way algorithmic address mapping described in this document MUST be implemented on all NAT66 devices. This mapping consists of mapping an internal IPv6 prefix, possibly a ULA, to/from an external prefix, typically a globally-routable unicast address, and making a complementary modification to the local IPv6 addresss. The same transformation is performed in both the inbound and outbound directions, so the only state that is needed on the NAT66 box to peform this transformation is knowledge of the internal and external address prefixes in use.

For the network shown in the first example diagram in the NAT66 Overview section above, we might have the following example:

Internal Prefix: FD01:0203:0405:/48 External Prefix: 2001:0DB8: 0001:/48

If a node with internal address FD01:0203:0405:0001::1234 sends an outbound packet through the NAT66 device, the resulting external address will be 2001:0DB8:0001:D550::1234. The resulting address is obtained by calculating the checksum of both the internal and external 48-bit prefixes, sutracting the internal prefix from the external prefix using one's complement arithmetic and adding the result to the 16-bit subnet field (in this case 0x0001).

To show the work:

The one's complement checksum of FD01:0203:0405 is 0xFCF5. The one's complement checksum of 2001:0DB8:0001 is 0xD245. Using one's complement math, 0xD245 - 0xFCF5 = 0xD54F. The subnet mask in the original packet is 0x0001. Using one's complement math, 0x0001 + 0xD54F = 0xD550. Since 0xD550 != 0xFFFF, it is not changed to 0x0000.

So, the value 0xD550 is written in the 16-bit subnet mask area, resulting in a mapped external address of 2001:0DB8:0001:D550::1234.

When a response packet is received, it will contain the destination address 2001:0DB8:0001:D550::0001, which will be mapped using the same mapping algorithm, back to FD01:0203:0405:0001::1234.

In this case, the difference between the two prefixes will be calculated as follows:

Using one's complement math, 0xFCF5 - 0xD245 = 0x2AB0. The subnet mask in the original packet = 0xD550. Using one's complement math, 0xD550 + 0x2AB0 = 0x0001. Since 0x0001 != 0xFFFF, it is not changed to 0x0000.

So the value 0x0001 is written into the subnet field, and the internal value of the subnet field is properly restored.

This mapping results in no modification of the Interface Identifier (IID), which is held in the lower half of the IPv6 address, so it will not interfere with future protocols that may use unique IIDs for node identification.

Use of this mapping is restricted to cases where both the internal and external prefixes are 48 bits long (a /48) or shorter, leaving at least 16 subnet bits that can be modified to ensure checksum neutrality. This may not be a significant limitation in practice, because it is expected that most NAT66 devices will be used to map between a provider-allocated external prefix of /48 or shorter and a ULA that uses the same prefix length as the external prefix. If necessary, however, there are several options for performing checksum correction in the IID portion of addresses that have a shorter prefix.

Furthermore, to avoid issues with one's complement arithmetic, subnet masks with all ones (0xFFFF) in bits 49-64 cannot be used on networks behind a NAT66 device.

7. Prefixes for Internal Addressing

NAT66 devices MUST support manual configuration of internal and external address prefixes, and MUST NOT place any restrictions on those prefixes except that they be valid IPv6 unicast address prefixes, as described in [RFC4291], and that they are /48 or shorter.

8. NAT Behavioral Requirements

NAT66 devices MUST support hairpinning behavior, as defined in the NAT Behavioral Requirements for UDP document [RFC4787]. This means that when a NAT66 device receives a packet on the internal interface that has a destination address that matches the site's external prefix, it will translate the packet and forward it internally. This allows internal nodes to reach other internal nodes using their external, global addresses when necessary.

Because NAT66 does not perform port mapping and uses a one-to-one, reversible mapping algorithm, it is not clear that any of the other NAT behavioral requirements apploy to NAT66. However, this topic should be discussed in more detail.

NAT66 devices that do not have a manually configured internal prefix SHOULD randomly generate a /48 ULA prefix for the internal network and advertise that prefix in router advertisements. NAT66 boxes with more than one internal interface SHOULD assign a (non-0xFFFF) subnet number to each link, and include the subnet number in router advertisements on the corresponding link. NAT66 devices that generate a ULA prefix MUST generate the prefix using a random number as described in RFC4291 [RFC4193], and SHOULD store the randomly generated prefix is non-volatile storage for continued use.

9. A Note on Port Mapping

In addition to overwriting IP addresses when packets are forwarded, NAT44 devices often overwrite the source port number in outbound traffic, and the destination port number in inbound traffic. This mechanism is called "port mapping".

The major benefit of port mapping is that it allows multiple computers to share a single IPv4 address. A large number of internal IPv4 addresses (typically from the 10.0.0.0/8 prefix) can be mapped into a single external, globally routable IPv4 address, with the local port number used to identify which internal node should receive each inbound packet. This address amplification feature should not

be needed in IPv6, where every attached network should be assigned at least a /48 prefix, leaving room for 16 subnet bits and a 64 bit Interface Identifier [RFC3587].

Since port mapping requires re-writing a portion of the transport layer header, it requires NAT66 devices to be aware of all of the transport protocols that they forward, thus stifling the development of new and improved transport protocols. Modifying the transport layer header is incompatible with security mechanisms that encrypt the full IP payload, and restricts the NAT66 device to forwarding transport layers that use weak checksum algorithms that are easily recalculated in routers. Since there is significant detriment caused by modifying transport layer headers and very little, if any, benefit to the use of port mapping in IPv6, NAT66 devices that comply with this specification MUST NOT perform port mapping.

10. SAF Considerations

There should be some discussion of how this document relates to the requirements listed in the IPv6 SAF Considerations draft [I-D.thaler-ipv6-saf]. Because NAT66 defines a fully reversible mapping, it would only be necessary for a host to get access to the external prefix that is being used for communication in order to perform a local mapping. However, it is not clear how that could be accomplished while maintaining the requirement for Address Independence.

11. Security Considerations

When NAT66 is deployed using the two-way, algorithmic address mapping defined in the document, it allows direct inbound connections to internal nodes. While this can be viewed as a benefit of NAT66 vs. NAT44, it does open internal nodes to attacks that would not be possible in a NAT44 network. Although this situation is not substantially worse, from a security standpoint, than running IPv6 with no NAT, some enterprises may assume that a NAT66 device will offer similar protection to a NAT44 device. For this reason, it is RECOMMENDED that NAT66 devices include an IPv6 firewall function, and the firewall function SHOULD be configured by default to block all incoming connections. Administrators could then enable inbound connectivity for specific ports by reconfiguring the firewall.

12. IANA Considerations

This document has no IANA considerations.

13. Acknowledgements

The checksum-neutral algorithmic address mapping described in this document is based on e-mail written by Iljtsch Van Beijnum. A similar mapping mechanism to the one described in this document was previously described in a document that can be found here: http://users.piuha.net/chvogt/pub/2008/vogt-2008-six-one-router-design.pdf. [TBD, move to an informative reference].

The following people provided advice or review comments that substantially improved this document: Jari Arrko, Iljtsch Van Beijnum, Remi Depres, Tony Hain, Ed Jankiewicz, Dave Thaler, Mark Townsley.

This document was written using the xml2rfc tool described in RFC 2629 [RFC2629].

14. Change Log

14.1. Changes Between -00 and -01

There were several minor changes made between the -00 and -01 versions of this draft:

- Added Fred Baker as a co-author.
- Minor mathematical corrections.
- Added AH to paragraph on NAT security issues.
- Added additional NAT topologies to overview (diagrams TBD).

14.2. Changes between -01 and -02

There were further changes made between -01 and -02:

- Removed topology hiding mechanism.
- Added diagrams.
- Made minor updates based on mailing list feedback.
- Added discussion of IPv6 SAF document.
- Added applicability section.

- Added discussion of Address Independence requirement.

- Added hairpining requirement and discussion of applicability of other NAT behavioral requirements.

Note: There were also some more major mailing list comments, but it hasn't been clear where consensus lies on a lot of the issues that were raised. So, changes of that magnitude will probably have to wait until there is a WG to reach consensus and chairs to determine when it has been reached.

15. References

15.1. Normative References

[RFC1071] Braden, R., Borman, D., Partridge, C., and W. Plummer, "Computing the Internet checksum", RFC 1071, September 1988.

[RFC1624] Rijsinghani, A., "Computation of the Internet Checksum via Incremental Update", RFC 1624, May 1994.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3587] Hinden, R., Deering, S., and E. Nordmark, "IPv6 Global Unicast Address Format", RFC 3587, August 2003.

[RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", RFC 4193, October 2005.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

[RFC4787] Audet, F. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, January 2007.

15.2. Informative References

[I-D.thaler-ipv6-saf] Thaler, D., "Source Address Finding (SAF) for IPv6 Translation Mechanisms", draft-thaler-ipv6-saf-01 (work in progress), February 2009.

[RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, June 1999.

[RFC2993] Hain, T., "Architectural Implications of NAT", RFC 2993, November 2000.

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

Authors' Addresses

Margaret Wasserman
Sandstorm Enterprises
14 Summer Street
Malden, MA 02148
USA

Phone: +1 781 333 3200
Email: mrw@lilacglade.org
URI: http://www.sandstorm.net

Fred Baker
Cisco Systems
Santa Barbara, California 93117
USA

Phone: +1-408-526-4257
Email: fred@cisco.com