

Network Working Group  
Internet-Draft  
Intended status: Informational  
Expires: January 12, 2012

D. Zhang  
X. Xu  
Huawei Technologies Co.,Ltd  
M. Boucadair  
France Telecom  
July 11, 2011

Considerations on NAT64 Load-Balancing  
draft-zhang-behave-nat64-load-balancing-03

## Abstract

This document investigates several load-balancing approaches for NAT64 devices and analyzes the advantages and disadvantages of various prefix selection policies. Both stateless and stateful NAT64 schemes are considered in this document.

## Requirements Language

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](#)].

## Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on January 12, 2012.

## Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents

(<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

<a href="#">1.</a>	<a href="#">Introduction . . . . .</a>	<a href="#">3</a>
<a href="#">2.</a>	<a href="#">Terminology . . . . .</a>	<a href="#">3</a>
<a href="#">3.</a>	<a href="#">Reminder on the Load Balancing Objectives . . . . .</a>	<a href="#">3</a>
<a href="#">3.1.</a>	<a href="#">Inbound Load Balancing . . . . .</a>	<a href="#">4</a>
<a href="#">3.2.</a>	<a href="#">Outbound Load Balancing . . . . .</a>	<a href="#">5</a>
<a href="#">4.</a>	<a href="#">Basic Load Balancing Considerations . . . . .</a>	<a href="#">5</a>
<a href="#">5.</a>	<a href="#">Stateless NAT64 . . . . .</a>	<a href="#">6</a>
<a href="#">5.1.</a>	<a href="#">Anycast-based Mode . . . . .</a>	<a href="#">6</a>
<a href="#">5.2.</a>	<a href="#">DHCPv6-based Mode . . . . .</a>	<a href="#">7</a>
<a href="#">5.3.</a>	<a href="#">NAT64 Farm . . . . .</a>	<a href="#">7</a>
<a href="#">6.</a>	<a href="#">Stateful NAT64 . . . . .</a>	<a href="#">7</a>
<a href="#">6.1.</a>	<a href="#">Anycast-based Mode . . . . .</a>	<a href="#">8</a>
<a href="#">6.2.</a>	<a href="#">Prefix64 Selection Policy . . . . .</a>	<a href="#">8</a>
<a href="#">6.2.1.</a>	<a href="#">Source-Based Prefix Selection Policy . . . . .</a>	<a href="#">8</a>
<a href="#">6.2.2.</a>	<a href="#">Destination-Based Prefix Selection Policy . . . . .</a>	<a href="#">9</a>
<a href="#">6.2.3.</a>	<a href="#">Round-Robin Prefix Selection Policy . . . . .</a>	<a href="#">10</a>
<a href="#">6.2.4.</a>	<a href="#">Dynamic Prefix Selection Policy . . . . .</a>	<a href="#">10</a>
<a href="#">6.3.</a>	<a href="#">Options for Implementing Load-balancers . . . . .</a>	<a href="#">11</a>
<a href="#">6.3.1.</a>	<a href="#">DNS64 Servers . . . . .</a>	<a href="#">11</a>
<a href="#">6.3.2.</a>	<a href="#">Prefix64 Assigners . . . . .</a>	<a href="#">12</a>
<a href="#">7.</a>	<a href="#">IANA Considerations . . . . .</a>	<a href="#">13</a>
<a href="#">8.</a>	<a href="#">Security Considerations . . . . .</a>	<a href="#">13</a>
<a href="#">9.</a>	<a href="#">Contributors . . . . .</a>	<a href="#">14</a>
<a href="#">10.</a>	<a href="#">References . . . . .</a>	<a href="#">15</a>
<a href="#">10.1.</a>	<a href="#">Normative References . . . . .</a>	<a href="#">15</a>
<a href="#">10.2.</a>	<a href="#">Informative References . . . . .</a>	<a href="#">15</a>
	<a href="#">Authors' Addresses . . . . .</a>	<a href="#">15</a>

## 1. Introduction

NAT64 devices [[I-D.ietf-behave-v6v4-xlate-stateful](#)] are facilities deployed on the boundaries between IPv6 and IPv4 networks to facilitate the communication between IPv6-only clients and IPv4 servers.

This document proposes several load-balancing approaches for NAT64 devices, which can be utilized in the delivery of highly available services, and compares the advantages and disadvantages of various prefix selection policies.

The issues with failover and redundancy are outside the scope of this document. An in depth analysis of those issues is elaborated in [[I-D.xu-behave-stateful-nat-standby](#)].

## 2. Terminology

This document makes use of the terms defined in [I-D.ietf-behave-v6v4-xlate-stateful], [[I-D.ietf-behave-dns64](#)] and [[RFC6052](#)]. Below are provided the terms specific to this document:

- o Prefix64: an IPv6 prefix used for synthesizing IPv6 addresses representing IPv4 hosts in the IPv6 realm. See [[RFC6052](#)] for more details on how IPv4-embedded IPv6 addresses are built.
- o NAT64-enabled device (or NAT64 device for short): is a device which embeds a NAT64 function as defined in [I-D.ietf-behave-v6v4-xlate-stateful].
- o A load-balancer is a facility which can select a NAT64 device from a set of deployed devices for a given IPv6-only client according to the pre-specified policy. Typically, a load-balancer provisions a client with an IPv6 address of the IPv4-only server that the client is going to access. Prefix64 is elaborately

selected by the load-balancer so that the corresponding NAT64 device can intercept the packets between the above communicating parties and correctly process them.

### [3.](#) Reminder on the Load Balancing Objectives

Load balancing is a technique used by network operators (including service providers, enterprise networks, etc.) to distribute the load among several ingress/egress points, several paths, several topologies, etc.

Zhang, et al.

Expires January 12, 2012

[Page 3]

---

Internet-Draft

NAT64 Load Balancing

July 2011

In the context of IPv4-IPv6 interconnection, load balancing is mainly motivated by the needs listed below:

- o to optimize the resources usage of deployed NAT64 devices (e.g., several ingress/egress NAT64 devices);
- o to avoid congesting a single NAT64 device while other free resources are still available;
- o to optimize IPv6-IPv4 interconnection costs especially when several NAT64 providers are involved.

Techniques to balance the load among a set of NAT64 devices can be achieved on a load-balancer managing a farm of NAT64 devices, on a single NAT64 device to select the appropriate outbound interface, or it can be implicitly achieved owing to dedicated tweaking operations (e.g., use of anycast-based service, use of distinct Prefix64, etc.). Note that considerations to balance the traffic between several outbound interfaces of a NAT64 device are out of scope of this document.

Various types of load balancing can be considered as defined in the following sub-sections.

#### [3.1.](#) Inbound Load Balancing

Inbound load balancing means that incoming traffic (i.e., the traffic received from an IPv4-only host) is distributed among a set of NAT64 devices located at the boundary of the IPv4 realm and the IPv6 one.

In a stateful NAT64 case, inbound load balancing cannot be explicitly configured because IPv4-only clients can not initiate sessions to an IPv6-only server (except for IPv6-only hosts which pre-installed static entries in the NAT64 using PCP [[I-D.ietf-pcp-base](#)] for instance).

In a stateless NAT64 case, inbound load balancing can be achieved by configuring distinct IPv4 address pools on each stateless NAT64 device (or these pools are to be configured with distinct routing metrics in each NAT64 device). This practice may lead to asymmetric paths (i.e., distinct NAT64 devices will handle the outgoing and the inbound packets) if the same NSP (Network Specific Prefix, [[RFC6052](#)]) is provisioned on those NAT64 devices. This can be seen as an issue for some operators because the legal stored data and activity logs can be increased. If downstream and upstream paths have similar characteristics (e.g., one-way delay, one-way delay variation, throughput), the path asymmetry is not an issue from a service perspective.

Note: As an alternative to address this problem, a load balancer needs to be deployed on each side of the NAT64 devices. The balancers need to know the policies of each other so as to work in a cooperative way. Refer to [Section 6.2](#) for more discussion.

### [3.2](#). Outbound Load Balancing

Outbound load balancing means the effort of distributing outgoing traffic (i.e., the traffic issued by IPv6-only hosts) among a set of NAT64 devices.

Only this flavor of load balancing is elaborated in the remainder of this document.

## [4](#). Basic Load Balancing Considerations

In practice, it is important to guarantee the outgoing traffic and the associated incoming traffic is stuck to a same stateful NAT64 device, so that the NAT64 device can get essential knowledge to correctly process the incoming packets. That is, if a stateful NAT64 device processes a request from an IPv6 client to an IPv4 server, it

MUST be able to intercept and process the correspondent reply from the server. To achieve this, distinct external IPv4 addresses SHOULD be configured on each stateful NAT64 device. Otherwise, if the same IPv4 address pool is provisioned on two distinct NAT64 devices (e.g., NAT64-A and NAT64-B) and since the routing paths may not be symmetric, outbound packets may be intercepted by NAT64-A while inbound packets may be received by NAT64-B. NAT64-B will reject the packets it received because no state to process these packets was instantiated beforehand, and thus the communication will fail.

The load balancing SHOULD NOT be solely done based on the traffic load distribution but SHOULD persevere the assignment of the same external IPv4 address for all active sessions initiated by an IPv6-only host. The criteria for distributing customers among a set of NAT64 devices may be implemented during the IPv6 configuration phase of a IPv6-only host or during the processing of actual traffic issued by that host.

In the circumstances where the static mapping entries of an IPv6-only client are pre-installed in a given stateful NAT64 device, the enforced load-balancing technique SHOULD "redirect" the traffic from the client to the NAT64-device where its static mappings are pre-installed.

Note: Some dynamic protocols such as PCP [[I-D.ietf-pcp-base](#)] may include manes to detect the unavailability of a NAT64 and to re-

install the mappings in the new discovered NAT64. But, for the manually configured mappings, the issue is still there.

From the operator's perspective, a load balancing solution SHOULD be deterministic, that is, that the actual traffic distribution should be strictly compliant with what is expected by the system manager. Furthermore, the operations of distributing the load among multiple NAT64 devices SHOULD be covered from end-users. This means that end-users should not be aware of the presence of multiple NAT64 devices in the core network and the selection of the appropriate NAT64 device should not assume any intervention by the customer/host.

When implementing load balancing, it should not lead to (severe) QoS degradation between potential paths. Note that the perceived quality may not only depend on the load balancing technique to distribute the

traffic among available path/nodes but it is closely related to the underlying topology (i.e., location of the NAT64 devices, routing metrics configuration, etc.).

An efficient load balancing system SHOULD NOT redirect the traffic to a congested NAT64 device while other NAT64 resources are available. Load indicators (i.e., the data reflecting the load imposed on NAT64 devices) may be disseminated to drive the process of selecting a NAT64 device to handle an ongoing IPv6 packet. These indicators may be based on (almost) real-time measurement tools or based on a traffic logic configured on the load-balancer (e.g., a NAT64 device can handle N IPv6-only hosts).

## [5.](#) Stateless NAT64

According to [[RFC6052](#)], IPv4-Translated and IPv4-Converted IPv6 addresses SHOULD use the same Network Specific Prefix (NSP). To distribute the traffic among a set of stateless NAT64 devices, the alternatives described hereafter can be envisaged. For the anycast-based mode the same Prefix64 is used while for the remaining options, specific Prefix64s are used.

### [5.1.](#) Anycast-based Mode

The same IPv6 NSP is provisioned to all stateless NAT64 devices; IPv6 hosts are distributed natively among several stateless NAT64 devices. This means that the closest (from a routing perspective) stateless NAT64 device will be used to process IPv6 (resp. IPv4) packets destined to an IPv4 (resp. IPv6) destination.

The efficiency of this mode largely depends on the underlying topology (e.g., location of NAT64 devices) and routing engineering

policies. Moreover, a stateless NAT64 device may be overloaded if the routing is not appropriately tuned and/or if the NAT64 devices are not appropriately dimensioned.

The introduction or the removal of NAT64 device(s) can be achieved without modifying the configuration of DHCPv6 servers. During failure events of a NAT64 system, other NAT64 devices can handle the traffic without any intervention.

### [5.2.](#) DHCPv6-based Mode

To implement this mode, each stateless NAT64 device is configured with a dedicated NSP. During the IPv6 prefix assignment phase, requesting IPv6 hosts are provided with IPv4-Translated IPv6 prefix using the NSP of the NAT64 device that will be used to handle traffic issued from those hosts and destined to an IPv4 host.

If DHCPv6 is used for the provisioning of IPv6 prefixes, DHCPv6 servers SHOULD be provided with the number of customers to be serviced per dedicated NSP (i.e., an NSP prefix identifies a stateless NAT64 device). Dynamic load information (based on real time monitoring) MAY be provided to the DHCPv6 to drive the process of IPv6 prefix assignment and for better utilization of available NAT64 resources. Furthermore, and for routing optimization purposes and for service stability purpose (e.g., use the same NAT64 device hosting PCP-instructed port forwarding entries), other topological information SHOULD be used to tag the customers that should be serviced by each NAT64 device.

### [5.3.](#) NAT64 Farm

An additional scheme would be the deployment of a farm of NAT64 devices with a load-balancer which is responsible for redirecting the traffic to the appropriate NAT64 instance. Unlike stateful NAT64, both IPv4 and IPv6 flows can be load balanced.

In this scenario, the same NSP SHOULD be used for all NAT64 devices belonging to the same farm.

Techniques to distribute the load among the NAT64 devices of the farm are similar to load-balancing techniques among several outbound interfaces of the same NAT64 system.

## [6.](#) Stateful NAT64

Two variant of the load balancing techniques are elaborated hereafter. Unlike the first mode, anycast-based, the second category



are provided below.

### [6.1.](#) Anycast-based Mode

This mode assumes that the same IPv6 prefix (i.e., NSP or WKP, see [\[RFC6052\]](#)) is provisioned to all deployed stateful NAT64 devices. DNS64 function is provisioned with that prefix used for synthesizing AAAA records.

As stated in [Section 4](#), distinct IPv4 address pools are configured to each NAT64 device. This ensures path symmetry; which means that the same NAT64 device will be used for handling both outbound and inbound packets exchanged in a same stateful conversation between two hosts.

The distribution of the traffic among deployed NAT64 devices is natively achieved relying on the underlying routing configuration. Off-line traffic engineering tools can be used to appropriately tweak the routing metrics so as to allow for acceptable traffic distribution.

The same NAT64 device SHOULD be used to handle all the packets issued by a given IPv6 host so that the same external IPv4 address is used to represent that host in the IPv4 realm. This means that oscillation phenomena induced by underlying routing MUST be avoided. By oscillation it is meant that the traffic customer is balanced between two NAT64 devices. The routing oscillation can be avoided owing to (off-line/on-line) traffic engineering techniques to select the appropriate location of the NAT64 devices in the network, the setting of underlying routing weights, establishment of explicit MPLS LSPs, etc.

### [6.2.](#) Prefix64 Selection Policy

It is RECOMMENDED that the functionality of load balancers should be integrated into dedicated servers. Therefore, load-balancing can be transparent for IPv6-only hosts. The design options of load balancers are discussed in [Section 6.3](#).

The following sub-sections elaborate on various modes for the prefix selection.

#### [6.2.1.](#) Source-Based Prefix Selection Policy

A source-based prefix selection policy allows a load-balancer to select Prefix64s according to the IPv6 addresses of its IPv6-only clients. For instance, when using a source-based prefix selection policy, the load-balancer in the above example can allocate an IPv6

address with Prefix64-A for the IPv4-only server if the IPv6 address of the client is odd, and Prefix64-B otherwise.

#### [6.2.1.1.](#) Pros

It is simple and has enough entropy to ensure reasonable load balancing across different NAT64 devices. 2.

The users are consistently assigned to the same NAT64 device for every outbound session. This is important because some applications identify a unique user across multiple transactions using the source IP address; examples include FTP and SSL VPNs. In addition, it is easier for a network management system (NMS) to monitor and manage the activities of a user. For instance, a NMS can collect the information about number of the concurrent sessions initiated by a user from a single NAT64 device. However, when using other policies, a user is not stuck to a NAT64 device, and thus NMS may have to collect such information from multiple NAT64 devices.

#### [6.2.1.2.](#) Cons

The efficiency of this procedure depends on the selection criteria and may not be deterministic in some cases where the traffic may be redirected to a congested NAT64 device.

### [6.2.2.](#) Destination-Based Prefix Selection Policy

A destination-based prefix selection policy requires a load-balancer to choose Prefix64s according to the IP addresses of the IPv4 targets. For instance, when using a destination-based prefix selection policy, the load-balancer in the above example can allocate an IPv6 address with Prefix64-A for the IPv4 server if the IPv4 address of the server is odd, and prefix64-B otherwise. In practice, this type of policy can have lots of variations. For instance, when a DNS server is utilized as a load balancer, the server can select a prefix64 according to the hash value of the FQDN (Fully Qualified Domain Name) of the target server.

#### [6.2.2.1.](#) Pros

It is simple to implement;

#### [6.2.2.2.](#) Cons

A user accessing multiple IPv4 servers may be represented by multiple public IPv4 addresses since its traffic may be processed by different

NAT64 systems. This will cause authentication problems in the applications (e.g., FTP and SSL VPNs) which take advantage of the

source IP addresses to identify users across different sessions. 2.

A user can not be redirected to the NAT64 device where it has instructed its port forwarding entries; 3.

Since there are more users than the servers providing contents, there is not enough entropy to ensure good load balancing. The NAT64 device that services a popular web-site will have to undertake much traffic. It is possible to define some strategies to make major sites evenly assigned to different NAT64s, e.g.- Google to NAT64-A, Facebook to NAT64-B, However, this solution can be onerous and requires heavy human involvement.

#### [6.2.3.](#) Round-Robin Prefix Selection Policy

A round-robin prefix selection policy allows a load-balancer to use various Prefix64s circularly in different requesting sessions. For instance, in the above example, the load-balancer can choose Prefix64-A in the first requesting session, choose Prefix64-B in the second, choose Prefix64-A in the third, choose Prefix64-B in the fourth, and so on.

##### [6.2.3.1.](#) Pros

Ensures reasonable distribution among a set of NAT64 devices.

##### [6.2.3.2.](#) Cons

A given IPv6-only hosts may be redirected to distinct NAT64 devices. Therefore, distinct IPv4 address may be used to represent the IPv6-only host in the IPv4 realm;

A user can not be redirected to the NAT64 device where it has instructed its port forwarding entries;

Requires a load balancer (e.g., a DNS64 or a DHCP server) to keep track of the assignments.

#### [6.2.4.](#) Dynamic Prefix Selection Policy

Although the capability of NAT64 devices can be considered as a factor in the designing of the above three types of policies, they are still static and not able to be adjusted according the load changes of NAT64 devices in a timely fashion. If we intend to enable a load-balancer to dynamically modify its policies according to NAT64s' real-time load changes, a dynamic prefix selection policy is necessary. For instance, a DNS64 system or DHCPv6 can use SNMP to collect the information of the overheads (e.g., CPU utilizing rates,

free memory amounts, concurrent session numbers, and session numbers per second) imposed on different NAT64-based devices. Based on such information, the load-balancer can distribute the loads on different NAT64 devices in a more reasonable way.

#### [6.2.4.1.](#) Pros

This type of policy can effectively avoid the unbalanced distribution of overheads on different NAT64 devices.

#### [6.2.4.2.](#) Cons

Such a policy may introduce additional communication and management complexities to a NAT64 device. The complexity depends on the means used to disseminate the NAT64 load.

### [6.3.](#) Options for Implementing Load-balancers

In practice, the functionality of a load-balancer can be performed by, e.g.- a DNS64 server, a DNS server, a DHCP server, an edge router, or even an IPv6 host itself.

#### [6.3.1.](#) DNS64 Servers

When collaborating with NAT64 devices, a DNS64 server can be solicited by an IPv6-only client to initiate communications to an IPv4-only server identified by a FQDN.

Let us assume there is an IPv6-only client connected to an IPv6 network which attempts to communicate to an IPv4-only server. For the purpose of load balancing, the DNS64 server needs to select a Prefix64 based on one of the prefix selection policies defined in

[Section 6.2](#) and use it when synthesizing AAAA RRs. Using the synthesized IPv6 address, the IPv6-only client will take advantage of the NAT64 associated with the Prefix64 to communicate with the IPv4-only server.

When DNS64 is used as a means to load balance the hosts among a group of NAT, DNS64 SHOULD be able to assign the same NAT64 to the same hosts. Means to identify the host SHOULD be supported. This is not natively supported by DNS servers.

A drawback of this mode is that for traffic which does not require a DNS resolution, the packets may flow using a distinct NAT device, and therefore use a distinct external IP address.

### [6.3.2.](#) Prefix64 Assigners

[I-D.korhonen-behave-nat64-learn-analysis] analyzes various solutions for a host in an IPv6-only network to obtain the Prefix64 of a NAT64 device. With the Prefix64, the hosts can synthesize an appropriate IPv6 address which can route packets to the translator. In the designing of load balancers for NAT64 devices, these approaches are worthwhile to consider.

#### [6.3.2.1.](#) DNS64 Servers

In [[I-D.korhonen-behave-nat64-learn-analysis](#)], a NAPTR RR to represent NAT64's Prefix64 is analyzed as part of the candidate solutions. When using DNS servers to act as load balancers for NAT64 devices, multiple NAPTR RRs need to be added to the zone file. Every NAPTR RR consists of a Prefix64. Upon receiving a NAPTR query, the DNS server replies the requester with a NAPTR RR according to a pre-specified selection policy. Note that the destination-based prefix selection policy is not applicable in this case because the DNS server may lack the knowledge of the IP address of the queried IPv4 host.

#### [6.3.2.2.](#) DHCPv6 Servers

It is mentioned in [[I-D.korhonen-behave-nat64-learn-analysis](#)] that a

DHCPv6 server can be used to allocate Prefix64s for hosts, and so a DHCP server has a potential to act as a load balancer for NAT64 devices. Similar with the solution proposed in [Section 6.3.2.1](#), it is difficult for a DHCP server to identify the IP addresses of the IPv4 hosts which its clients intend to communicate with. Therefore, only the source-based policy, the round-robin policy, or the dynamic policy can be used in this approach.

Also, a DHCPv6 server can be adopted to allocate different DNS64 servers for its users in various standard DHCPv6 host configuration processes according to certain selection policies. Unlike the DNS64 servers discussed in [Section 6.3.1](#), in this case a DNS64 server needs to only synthesize AAAA records using a single Prefix64.

The load of NAT devices may be provided to DHCP servers to assist the selection of the DNS64 to be used for new connecting hosts.

#### [6.3.2.3](#). Default Gateways

[I-D.korhonen-behave-nat64-learn-analysis] also discusses the possibility of using Router Advertisement (RA) messages to transfer Prefix64s for IPv6 users. If the edge router is attached to only one multicast link, no prefix selection policy defined in [Section 6.2](#) can

be used. If the edge router is attached to multiple multicast links, the source-based policy, the round-robin policy or the dynamic policy can be used. Because at this phase it is difficult for an edge router to identify the IP addresses of the IPv4 hosts which the IPv6 hosts will communicate with, the destination-based prefix selection policy is unfeasible.

#### [6.3.2.4](#). IPv6 Clients

It is possible for an IPv6 host to learn multiple Prefix64s through the approaches defined in [[I-D.korhonen-behave-nat64-learn-analysis](#)] and then select one based on a certain prefix selection policy. Such a policy can be the destination-based policy, the source-based policy (only one prefix64 is used), the round-robin policy or the dynamic policy.

This solution is not deterministic and can lead to congesting a given NAT64 device.

## [7.](#) IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

## [8.](#) Security Considerations

As mentioned previously, all the traffic between an IPv6 host and an IPv4 host should be intercepted and processed by a same NAT64 device. However, when using certain policies (e.g., the destination-based prefix selection policy and the Round-Robin prefix selection policy), this requirement cannot be fulfilled. The traffic of a user will be distributed to different NAT64 devices. Under such a circumstance, it may be difficult for network management systems to collect information from different NAT64 devices in order to monitor users' behavior in a real in time fashion. In addition, it can be difficult for intrusion detection/prevision systems to combine the operations of a user so as to reason whether she is trying to perform a multi-step attack.

Another security concern is load balancers. Because load balancers play an important role in distributing traffic to different NAT64 devices, the communication between users and load balancers should be secured. Otherwise, attackers may disturb load balancing and carry out DDoS attacks by modifying the packets sent from load balancers.

## [9.](#) Contributors

The following individuals have contributed to this document:

Xuwei Wang  
Huawei Technologies Co.,Ltd  
KuiKe Building, No.9 Xinxu Rd.,  
Hai-Dian District, Beijing 100085  
P.R. China

Email: wangxuwei@huawei.com

Yan Wang  
CNNIC  
No.4 South 4th Street,  
Beijing, Zhongguancun 100190  
P. R. China

Email: wangyan-lab@cnnic.cn

Cameron Byrne  
T-Mobile USA  
3617 131st Ave SE  
Bellevue, WA 98006  
US  
Email: cameron.byrne@t-mobile.com

Dong Zhang  
Huawei Symantec  
KuiKe Building, No.9 Xinxu Rd.,  
Beijing, Hai-Dian District 100085  
P. R. China

Email: zhangdong\_rh@huaweisymantec.com

Zhenqiang Li  
China Mobile  
Unit2, Dacheng Plaza, No. 28 Xuanwumenxi Ave, Xicheng District  
Beijing 100053  
P.R. China

Email: lizhenqiang@chinamobile.com

## [10.](#) References

Zhang, et al.	Expires January 12, 2012	[Page 14]
---------------	--------------------------	-----------

---

Internet-Draft	NAT64 Load Balancing	July 2011
----------------	----------------------	-----------

### [10.1.](#) Normative References

[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-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.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", [RFC 6052](#), October 2010.

## [10.2.](#) Informative References

- [I-D.ietf-pcp-base]  
Wing, D., Cheshire, S., Boucadair, M., Penno, R., and P. Selkirk, "Port Control Protocol (PCP)",  
[draft-ietf-pcp-base-13](#) (work in progress), July 2011.
- [I-D.korhonen-behave-nat64-learn-analysis]  
Korhonen, J. and T. Savolainen, "Analysis of solution proposals for hosts to learn NAT64 prefix",  
[draft-korhonen-behave-nat64-learn-analysis-02](#) (work in progress), February 2011.
- [I-D.xu-behave-stateful-nat-standby]  
Xu, X., Boucadair, M., Lee, Y., and G. Chen, "Redundancy Requirements and Framework for Stateful Network Address Translators (NAT)",  
[draft-xu-behave-stateful-nat-standby-06](#) (work in progress), October 2010.

Authors' Addresses

Dacheng Zhang  
Huawei Technologies Co.,Ltd  
KuiKe Building, No.9 Xinxu Rd.,  
Hai-Dian District, Beijing 100085  
P.R. China

Email: zhangdacheng@huawei.com

Xiaohu Xu  
Huawei Technologies Co.,Ltd  
KuiKe Building, No.9 Xinxu Rd.,  
Hai-Dian District, Beijing 100085  
P.R. China

Email: xuxh@huawei.com

Mohamed Boucadair  
France Telecom  
Rennes,  
France

Email: mohamed.boucadair@orange-ftgroup.com

