BEHAVE F. Audet, Ed.
Internet-Draft Nortel Networks
Expires: July 11, 2005 C. Jennings
Cisco Systems
January 10, 2005

NAT Behavioral Requirements for Unicast UDP
draft-ietf-behave-nat-udp-00

Status of this Memo

This document is an Internet-Draft and is subject to all provisions of section 3 of RFC 3667. By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she become aware will be disclosed, in accordance with RFC 3668.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

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

This Internet-Draft will expire on July 11, 2005.

Copyright Notice

Copyright (C) The Internet Society (2005).

Abstract

This document defines basic terminology for describing different types of NAT behavior when handling Unicast UDP, and defines a set of requirements that would allow many applications, such as multimedia communications or on-line gaming, to work consistently. Developing NATs that meet this set of requirements will greatly increase the likelihood that these applications will function properly.

Table of Contents

1. Applicability Statement

The purpose of this specification is to define a set of requirements for NATs that would allow many applications, such as multimedia communications or on-line gaming, to work consistently. Developing NATs that meet this set of requirements will greatly increase the likelihood that these applications will function properly.

The requirements of this specification apply generally to all NAT variations, including the ones described in RFC 2663 [3] (Traditional NAT, Basic NAT, NAPT, Bi-directional NAT, Twice NAT, and Multihomed NATs). However, it is not within the scope of this specification to address all issues specific to all possible NAT variations.

This document is meant to cover NATs of any size, from small residential NATs to large Enterprise NATs. However, it should be understood that Enterprise NATs normally provide much more than just NAT capabilities: for example, they typically provide Firewall capabilities. Firewalls is specifically out-of-scope of this specification. However, this specification does cover the inherent filtering aspects of NAT. Many large Enterprise NATs also have additional requirements on security, multihoming and so forth, which may impose further restrictions on the NAT capabilities. These extra requirements specifically targeted at large Enterprise NATs are outside the scope of this document. Furthermore, it is understood that certain NATs, especially NATs that have to satisfy additional requirements such as Firewall, may choose to be compliant to only certain requirements from this specification.

Approaches using directly signaled control off the middle boxes such as Midcom, UPnP, or in-path signaling are out of scope.

UDP Relays are out of the scope of this document.

Application aspects are out of scope as the focus is strictly on the NAT itself.

This document only covers the UDP Unicast aspects of NAT traversal and does not cover TCP, IPSEC, or other protocols. Since the document is for UDP only, packet inspection below the UDP layer (including RTP) is also out-of-scope.

2. Introduction

Network Address Translators (NAT) are well known to cause very significant problems with applications that carry IP addresses in the payload RFC 3027 [5]. Applications that suffer from this problem include Voice Over IP and Multimedia Over IP (e.g., SIP [6] and H.323

[19]), as well as online gaming.

Many techniques are used to attempt to make realtime multimedia applications, online games, and other applications work across NATs. Application Level Gateways [3] are one such mechanism. STUN [7] describes a UNilateral Self-Address Translation (UNSAF) mechanism [2]. UDP Relays have also been used to enable applications across NATs, but these are generally seen as a solution of last resort. ICE [16] describes a methodology for using many of these techniques and avoiding a UDP Relay unless the type of NAT is such that it forces the use of such a UDP Relay. This specification defines requirements for improving NATs. Meeting these requirements ensures that applications will not be forced to use UDP media relay.

Several recommendations regarding NATs for Peer-to-Peer media were made in [17] and this specification derives some of its requirements from that draft.

As pointed out in UNSAF [2], "From observations of deployed networks, it is clear that different NAT boxes' implementation vary widely in terms of how they handle different traffic and addressing cases." This wide degree of variability is one part of what contributes to the overall brittleness introduced by NATs and makes it extremely difficult to predict how any given protocol will behave on a network traversing NATs. Discussions with many of the major NAT vendors have made it clear that they would prefer to deploy NATs that were deterministic and caused the least harm to applications while still meeting the requirements that caused their customers to deploy NATs in the first place. The problem the NAT vendors face is they are not sure how best to do that or how to document how their NATs behave.

The goals of this document are to define a set of common terminology for describing the behavior of NATs and to produce a set of requirements on a specific set of behaviors for NATs. The requirements represent what many vendors are already doing, and it is not expected that it should be any more difficult to build a NAT that meets these requirements or that these requirements should affect performance.

This document forms a common set of requirements that are simple and useful for voice, video, and games, which can be implemented by NAT vendors. This document will simplify the analysis of protocols for deciding whether or not they work in this environment and will allow providers of services that have NAT traversal issues to make statements about where their applications will work and where they will not, as well as to specify their own NAT requirements.

3. 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 [1].

A NAT that complies with all of the mandatory requirements of this specification (i.e., the "MUST"), is "compliant with this specification." A NAT that complies with all of the requirements of this specification (i.e., including the "RECOMMENDED" and SHOULD) is "fully compliant with all the mandatory and recommended requirements of this specification."

Readers are urged to refer to RFC 2263 [3] for information on NAT taxonomy and terminology. Traditional NAT is the most common type of NAT device deployed. Readers may refer to RFC 3022 [4] for detailed information on traditional NAT. Traditional NAT has two main varieties - Basic NAT and Network Address/Port Translator (NAPT).

NAPT is by far the most commonly deployed NAT device. NAPT allows multiple internal hosts to share a single public IP address simultaneously. When an internal host opens an outgoing TCP or UDP session through a NAPT, the NAPT assigns the session a public IP address and port number so that subsequent response packets from the external endpoint can be received by the NAPT, translated, and forwarded to the internal host. The effect is that the NAPT establishes a NAT session to translate the (private IP address, private port number) tuple to (public IP address, public port number) tuple and vice versa for the duration of the session. An issue of relevance to peer-to-peer applications is how the NAT behaves when an internal host initiates multiple simultaneous sessions from a single (private IP, private port) endpoint to multiple distinct endpoints on the external network. In this specification, the term "NAT" refers to both "Basic NAT" and "Network Address/Port Translator (NAPT)".

This document uses the term "session" as defined in RFC 2663: "TCP/UDP sessions are uniquely identified by the tuple of (source IP address, source TCP/UDP ports, target IP address, target TCP/UDP Port)."

This document uses the term "address and port mapping" as the translation between an external address and port and an internal address and port. Note that this is not the same as an "address binding" as defined in RFC 2663.

RFC 3489 [7] defines a terminology for different NAT variations. In particular, it uses the terms "Full Cone", "Restricted Cone", "Port Restricted Cone" and "Symmetric" to refer to different variations of

NATs applicable to UDP only. This specification refers to specific individual NAT behaviors instead of using the Cone/Symmetric terminology. The relationship between the Cone/Symmetric terminology and the individual behaviors defined in this specification is described.

4. Network Address and Port Translation Behavior

This section describes the various NAT behaviors applicable to NAT.

4.1 Address and Port Mapping

When an internal endpoint opens an outgoing UDP session through a NAT, the NAT assigns the session an external IP address and port number so that subsequent response packets from the external endpoint can be received by the NAT, translated and forwarded to the internal endpoint. This is a mapping between an internal IP address and port IP:port and external IP:port tuple. It establishes the translation that will be performed by the NAT for the duration of the session. For many applications, it is important to distinguish the behavior of the NAT when there are multiple simultaneous sessions established to different external endpoints.

The key behavior to describe is the criteria for re-use of a mapping for new sessions to external endpoints, after establishing a first mapping between an internal X:x address and port and an external Y1:y1 address tuple. Let's assume that the internal IP address and port X:x is mapped to X1':x1' for this first session. The endpoint then sends from X:x to an external address Y2:y2 and gets a mapping of X2':x2' on the NAT. The relationship between X1':x1' and X2':x2' for various combinations of the relationship between Y1:y1 and Y2:y2 is critical for describing the NAT behavior. This arrangement is illustrated in the following diagram:

```
                                      E
   +------+                 +------+  x
   |  Y1  |                 |  Y2  |  t
   +--+---+                 +---+--+  e
      | Y1:y1            Y2:y2  |     r
      +----------+   +----------+     n
                 |   |                a
         X1':x1' |   | X2':x2'        l
              +--+---+-+
   ...........|   NAT  |...............
              +--+---+-+              I
                 |   |                n
             X:x |   | X:x            t
                ++---++               e
                |  X  |               r
                +-----+               n
                                      a
                                      l
```

The following address and port mapping behavior are defined:

External NAT mapping is endpoint independent:
The NAT reuses the port mapping for subsequent sessions initiated from the same internal IP address and port (X:x) to any external IP address and port. Specifically, X1':x1' equals X2':x2' for all values of Y2:y2. From an RFC 3489 NAT perspective, this is a "Cone NAT" where the sub-type is really based on the filtering behavior.

External NAT mapping is endpoint address dependent:
The NAT reuses the port mapping for subsequent sessions initiated from the same internal IP address and port (X:x) only for sessions to the same external IP address, regardless of the external port. Specifically, X1':x1' equals X2':x2' if, and only if, Y2 equals Y1. From an RFC 3489 NAT perspective, but not necessarily a filtering perspective, this is a "Symmetric NAT".

External NAT mapping is endpoint address and port dependent:
The NAT reuses the port mapping for subsequent sessions initiated from the same internal IP address and port (X:x) only for sessions to the same external and port. Specifically, X1':x1' equals X2':x2' if, and only if, Y2:y2 equals Y1:y1. From an RFC 3489 NAT perspective, but not necessarily a filtering perspective, this is a "Symmetric NAT".

It is important to note that these three possible choices make no difference to the security properties of the NAT. The security

properties are fully determined by which packets the NAT allows in and which it does not. This is determined by the filtering behavior in the filtering portions of the NAT.

Some NATs are capable of assigning IP addresses from a pool of IP addresses on the external side of the NAT, as opposed to just a single IP address. This is especially common with larger NATs. Some NATs use the external IP address mapping in an arbitrary fashion (i.e. randomly): one internal IP address could have multiple external IP address mappings active at the same time for different sessions. These NATs have an "IP address pooling" behavior of "Arbitrary". Some large Enterprise NATs use an IP address pooling behavior of "Arbitrary" as a means of hiding the IP address assigned to specific endpoints by making their assignment less predictable. Other NATs use the same external IP address mapping for all sessions associated with the same internal IP address. These NATs have an "IP address pooling" behavior of "Paired." NATs that use an "IP address pooling" behavior of "arbitrary" can cause issues for applications that use multiple ports from the same endpoint but do not negotiate IP addresses individually (e.g., some applications using RTP and RTCP).

4.2 Port Assignment

4.2.1 Port Assignment Behavior

This section uses the following diagram for reference.

```
                                      E
   +-------+               +-------+  x
   |  Y1   |               |  Y2   |  t
   +---+---+               +---+---+  e
       | Y1:y1          Y2:y2  |      r
       +---------+   +---------+      n
                 |   |                a
         X1':x1' |   | X2':x2'        l
              +--+---+--+
   ...........|   NAT   |...............
              +--+---+--+             I
                 |   |                n
       +---------+   +---------+      t
       | X1:x1          X2':x2 |      e
   +---+---+               +---+---+  r
   |  X1   |               |  X2   |  n
   +-------+               +-------+  a
                                      l
```

Some NATs attempt to preserve the port number used internally when assigning a mapping to an external IP address and port (e.g., x=x1=x2=x1'=x2', or more succinctly, a mapping of X:x to X':x). A basic NAT, for example, will preserve the same port and will assign a different IP address from a pool of external IP addresses in case of port collision (e.g. X1:x to X1':x and X2:x to X2':x). This is only possible as long as the NAT has enough external IP addresses. If the port x is already in use on all available external IP addresses, then the NAT needs to switch from Basic NAT to a Network Address and Port Translator (NAPT) mode (i.e., X'=X1'=X2' and x=x1=x2 but x1'!=x2', or a mapping of X1:x to X':x1' and X2:x to X':x2'). This port assignment behavior is referred to as "port preservation". It does not guarantee that the external port x' will always be the same as the internal port x but only that the NAT will preserve the port if possible.

A NAT that does not attempt to make the external port numbers match the internal port numbers in any case (i.e., X1:x to X':x1', X2:x to X':x2') is referred to as "no port preservation".

Some NATs use "Port overloading", i.e. they always use port preservation even in the case of collision (i.e., X'=X1'=X2' and x=x1=x2=x1'=x2', or a mapping of X1:x to X':x, and X2:x to X':x). These NATs rely on the source of the response from the external endpoint (Y1:y1, Y2:y2) to forward a packet to the proper internal endpoint (X1 or X2). Port overloading fails if the two internal endpoints are establishing sessions to the same external destination.

Most applications fail in some cases with "Port Overloading". It is

clear that "Port Overloading" behavior will result in many problems. For example it will fail if two internal endpoints try to reach the same external destination, e.g., a server used by both endpoints such as a SIP proxy, or a web server, etc.)

When NATs do allocate a new source port, there is the issue of which IANA-defined range of port to choose. The ranges are "well-known" from 0 to 1023, "registered" from 1024 to 49151, and "dynamic/private" from 49152 through 65535. For most protocols, these are destination ports and not source ports, so mapping a source port to a source port that is already registered is unlikely to have any bad effects. Some NATs may choose to use only the ports in the dynamic range; the only down side of this practice is that it limits the number of ports available. Other NAT devices may use everything but the well-known range and may prefer to use the dynamics range first or possibly avoid the actual registered ports in the registered range. Other NATs preserve the port range if it is in the well-known range. It should be noted that port 0 is reserved and must not be used.

4.2.2 Port Parity

Some NATs preserve the parity of the UDP port, i.e., an even port will be mapped to an even port, and an odd port will be mapped to an odd port. This behavior respects the RFC 3550 [8] rule that RTP use even ports, and RTCP use odd ports. Some NATs preserve the parity of the UDP port, i.e., an even port will be mapped to an even port, and an odd port will be mapped to an odd port. This behavior respects the RFC 3550 rule that RTP use even ports and RTCP use odd ports when the application takes a single port number as a parameter and derives the RTP and RTCP port pair from that number. RFC 3550 allows any port numbers to be used for RTP and RTCP if the two numbers are specified separately, for example using RFC 3605 [9]. However, some implementations do not include RFC 3605 and do not recognize when the peer has specified the RTCP port separately using RFC 3605. If such an implementation receives an odd RTP port number from the peer (perhaps after having been translated by a NAT), and then follows the RFC 3550 rule to change the RTP port to the next lower even number, this would obviously result in the loss of RTP. NAT-friendly application aspects are outside the scope of this document. It is expected that this issue will fade away with time, as implementations improve. Preserving the port parity allows for supporting communication with peers that do not support explicit specification of both RTP and RTCP port numbers.

4.2.3 Port Contiguity

Some NATs attempt to preserve the port contiguity rule of RTCP=RTP+1.

These NATs do things like sequential assignment, port reservation and so forth. Sequential port assignment assumes that the application will open a mapping for RTP first and then open a mapping for RTCP. It is not practical to enforce this requirement on all applications. Furthermore, there is a glare problem if many applications (or endpoints) are trying to open mapping simultaneously. Port reservation is also problematic since it is wasteful, especially considering that a NAT can not reliably distinguish between RTP over UDP and other UDP packets where there is no contiguity rule. For those reasons, it would be too complex to attempt to preserve the contiguity rule by suggesting specific NAT behavior, and it would certainly break the deterministic behavior rule.

In order to support both RTP and RTCP, it will therefore be necessary that applications follows rules to negotiate both RTP and RTCP separately, and account for the very real possibility that the RTCP=RTP+1 rule will be broken. As this is an application requirement, it is outside of the scope of this document.

4.3 Mapping Refresh Direction

NAT UDP mapping timeout implementations vary but include the timer's value and the way the mapping timer is refreshed to keep the mapping alive.

The mapping timer is defined as the time a mapping will stay active without packets traversing the NAT. There is great variation in the values used by different NATs.

Some NATs keep the mapping active (i.e., refresh the timer value) when a packet goes from the internal side of the NAT to the external side of the NAT. This is referred to as having a NAT Outbound refresh behavior of "True".

Some NATs keep the mapping active when a packet goes from the external side of the NAT to the internal side of the NAT. This is referred to as having a NAT Inbound Refresh Behavior of "True".

Some NATs keep the mapping active on both, in which case both properties are "True".

4.4 Mapping Refresh Scope

If the mapping is refreshed for all sessions on that mapping by any outbound traffic, the NAT is said to have a NAT Mapping Refresh Scope of "Per mapping". If the mapping is refreshed only on a specific session on that particular mapping by any outbound traffic, the NAT is said to have a "Per session" NAT mapping Refresh Scope.

5. Filtering Behavior

This section describes various filtering behaviors observed in NATs.

5.1 Filtering of Unsolicited Packets

When an internal endpoint opens an outgoing UDP session through a NAT, the NAT assigns a filtering rule for the mapping between an internal IP:port (X:x) and external IP:port (Y:y) tuple.

The key behavior to describe is what criteria are used by the NAT to filter packets originating from specific external endpoints.

External filtering is endpoint independent:
The NAT filters out only packets not destined to the internal address and port X:x, regardless of the external IP address and port source (Z:z). The NAT forwards any packets destined to X:x. In other words, sending packets from the internal side of the NAT to any external IP address is sufficient to allow any packets back to the internal endpoint. From an RFC 3489 filtering perspective, this is a "Full Cone NAT".

External filtering is endpoint address dependent:
The NAT filters out packets not destined to the internal address X:x. Additionally, the NAT will filter out packets from Y:y destined for the internal endpoint X:x if X:x has not sent packets to Y previously (independently of the port used by Y). In other words, for receiving packets from a specific external endpoint, it is necessary for the internal endpoint to send packets first to that specific external endpoint's IP address. From an RFC 3489 filtering perspective, this is a "Restricted Cone NAT".

External filtering is endpoint address and port dependent:
This is similar to the previous behavior, except that the external port is also relevant. The NAT filters out packets not destined for the internal address X:x. Additionally, the NAT will filter out packets from Y:y destined for the internal endpoint X:x if X:x has not sent packets to Y:y previously. In other words, for receiving packets from a specific external endpoint, it is necessary for the internal endpoint to send packets first to that external endpoint's IP address and port. From an RFC 3489 filtering perspective, this is either a "Port Restricted Cone NAT" or a "Symmetric NAT" as they both have the same filtering behavior.

5.2 NAT Filter Refresh

The time for which a NAT filter is valid can be refreshed based on packets that are inbound, outbound, or going either direction. In the case of "External Filtering" of "Address dependent" or "Address and port dependent" NATs, the scope of the refresh could include the filters for just the particular port and destination or for all the ports and destinations sharing the same external address and port on the NAT.

6. Relationship with Cone and Symmetric NAT Terminology

This section describes the relationship between the Network Address and Port and Filtering behaviors defined in this document, and the Cone/Symmetric NAT terminology described in RFC 3489.

RFC 3489 defines the following variations. They have been slightly paraphrased for emphasizing the mapping behavior and the filtering behavior.

Full Cone:

1. A full cone NAT is one where all requests from the same internal IP address and port are mapped to the same external IP address and port.
2. Furthermore, any external host can send a packet to the internal host, by sending a packet to the mapped external address.

Restricted Cone:

1. A restricted cone NAT is one where all requests from the same internal IP address and port are mapped to the same external IP address and port.
2. Unlike a full cone NAT, an external host (with IP address X) can send a packet to the internal host only if the internal host had previously sent a packet to IP address X.

Port Restricted Cone:

1. A port restricted cone NAT is one where all requests from the same internal IP address and port are mapped to the same external IP address and port.
2. The restriction includes port numbers. Specifically, an external host can send a packet, with source IP address X and source port P, to the internal host only if the internal host had previously sent a packet to IP address X and port P.

Symmetric:

1. A symmetric NAT is one where all requests from the same internal IP address and port, to a specific destination IP

address and port, are mapped to the same external IP address and port. If the same host sends a packet with the same source address and port, but to a different destination, a different mapping is used.

2. Furthermore, only the external host that receives a packet can send a UDP packet back to the internal host.

Unfortunately, this terminology defined in RFC 3489 has been the source of much confusion. This terminology does not distinguish between the mapping behavior (conditions 1 above) and the filtering behavior (conditions 2 above).

The inferred definition of "Cone NAT" is quite clear since the same definition is used for all variations of Cone NAT:

- o A cone NAT is one where all requests from the same internal IP address and port are mapped to the same address and port.

A "Cone NAT" therefore only refers to the Network Address and Port mapping behavior. This maps to the "External NAT mapping is endpoint independent" defined in this specification.

The terms "Full", "Restricted", "Port Restricted" refers to their filtering behavior. They map respectively to the "External filtering is endpoint independent", "External filtering is endpoint address dependent" and "External filtering is address and port dependent" behaviors.

However, the Symmetric NAT definition is more troublesome as it bundles together the mapping and the filtering definitions. Condition 1 of the Symmetric NAT definition is the NAT behavior and condition 2 is the filtering behavior. However, they are not necessarily dependent: we have observed NATs that will conform to condition (1) but not to (2). Using RFC 3489, this type of NAT would be detected as a "Cone NAT" since it uses condition (2). Using a different algorithm such as the one described in NATCHECK [20] which uses condition (1), the same NAT would be detected as a "Symmetric NAT". If the endpoint receiving the media has a permissive policy on accepting media, condition (2) is more appropriate, but if it has a restrictive policy, condition (1) is more appropriate. Some view the "real" definition of Symmetric NAT to be condition 1 while others believes it is condition 2.

It was found that many devices' behaviors do not exactly fit into the described variations. For example, a device could be symmetric from a filtering point of view and Cone from a NAT point of view. Other aspects of NATs are not covered by this terminology: for example, many NATs will switch over from basic NAT (preserving ports) to NAPT (mapping ports) in order to preserve ports when possible.

The relationship between the RFC 3489 and the behaviors described in this document is easier to describe in a table:

	External Filtering Behavior		
External NAT Mapping Behavior	Endpoint Independent	Endpoint Address Dependent	Endpoint Address/Port Dependent
Endpoint Independent	Full Cone	Restricted Cone	Port Restricted Cone
Endpoint Address Dependent	Symmetric~ (a)	Symmetric~ (a, 2)	Symmetric~ (a, b)
Endpoint Address /Port Dependent	Symmetric~ (1)	Symmetric (1, 2)	Symmetric~ (1, b)

Where:

1. Satisfies condition 1 for Symmetric NAT: "All requests from the same internal IP address and port to a specific destination address and port are mapped to the same external IP address and port. If a host sends a packet with the same source address and port to different destination addresses or ports, a different mapping is used for each."
2. Satisfies condition 2 for Symmetric NAT: "Furthermore, only the external host that receives a packet can send a UDP packet back to the internal host."

And:

a) This is a variation on condition (1), but where the destination port is not of any consequence.

b) This one is a variation on condition (2) which is more restrictive and not covered in the definition of Symmetric: "Furthermore, only packets originating from a port of the external host that has received packets already on that port will be forwarded."

If conditions (1) and (2), but not (b) are met, this is a Symmetric NAT as per the definition of RFC 3489. This is denoted as "Symmetric" in the table. Otherwise, the NAT is not quite Symmetric and is denoted as "Symmetric~". In some cases these Symmetric~ NATs are slightly more restrictive than a real Symmetric NAT, and in other cases they are more permissive.

7. Hairpinning Behavior

If two hosts (called X1 and X2) are behind the same NAT and exchanging traffic, the NAT may allocate an address on the outside of the NAT for X2, called X2':x2'. If X1 sends traffic to X2':x2', it goes to the NAT, which must relay the traffic from X1 to X2. This is referred to as hairpinning and is illustrated below.

```
  NAT
+----+ from X1:x1 to X2':x2'   +-----+ X1':x1'
| X1 |>>>>>>>>>>>>>>>>>>>>>>>>>>>>>--+---
+----+                         |  v  |
                               |  v  |
                               |  v  |
                               |  v  |
+----+ from X1':x1' to X2:x2   |  v  | X2':x2'
| X2 |<<<<<<<<<<<<<<<<<<<<<<<<<<<<<--+---
+----+                         +-----+
```

Hairpinning allows two endpoints on the internal side of the NAT to communicate even if they only use each other's external IP addresses and ports.

More formally, a NAT that supports hairpinning forwards packets originating from an internal address, X1:x1, destined for an external address X2':x2' that has an active mapping to an internal address X2:x2, back to that internal address X2:x2. Note that typically X1' is the same as X2'.

Furthermore, the NAT may present the hairpinned packet with either an internal or an external source IP address and port. The hairpinning NAT behavior can therefore be either "External source IP address and port" or "Internal source IP address and port". "Internal source IP address and port" may cause problems by confusing an implementation that is expecting an external IP address and port.

8. Application Level Gateways

Certain NATs have implemented Application Level Gateways (ALGs) for various protocols, including protocols for negotiating peer-to-peer UDP sessions.

Certain NATs have these ALGs turned on permanently, others have them turned on by default but let them be be turned off, and others have them turned off by default but let them be turned on.

NAT ALGs may interfere with UNSAF methods and must therefore be used

with extreme caution.

9. Deterministic Properties

The classification of NATs is further complicated by the fact that under some conditions the same NAT will exhibit different behaviors. This has been seen on NATs that preserve ports or have specific algorithms for selecting a port other than a free one. If the external port that the NAT wishes to use is already in use by another session, the NAT must select a different port. This results in different code paths for this conflict case, which results in different behavior.

For example, if three hosts X1, X2, and X3 all send from the same port x, through a port preserving NAT with only one external IP address, called X1', the first one to send (i.e., X1) will get an external port of x but the next two will get x2' and x3' (where these are not equal to x). There are NATs where the External NAT mapping characteristics and the External Filter characteristics change between the X1:x and the X2:x mapping. To make matters worse, there are NATs where the behavior may be the same on the X1:x and X2:x mappings but different on the third X3:x mapping.

Some NATs that try to reuse external ports flow from two internal IP addresses to two different external IP addresses. For example, X1:x is going to Y1:y1 and X2:x is going to Y2:y2, where Y1:y1 does not equal Y2:y2. Some NATs will map X1:x to X1':x and will also map X2:x to X1':x. This works in the case where the NAT mapping is address port dependent. However some NATs change their behavior when this type of port reuse is happening. The NAT may look like it has NAT mappings that are independent when this type of reuse is not happening but may change to Address Port Dependent when this reuse happens.

Any NAT that changes the NAT mapping or the External Filtering at any point in time under any particular conditions is referred to as a "non-deterministic" NAT. NATs that don't are called "deterministic".

Non-deterministic NATs generally change behavior when a conflict of some sort happens, i.e. when the port that would normally be used is already in use by another mapping. The NAT mapping and External Filtering in the absence of conflict is referred to as the Primary behavior. The behavior after the first conflict is referred to as Secondary and after the second conflict is referred to as Tertiary. No NATs have been observed that change on further conflicts but it is certainly possible that they exist.

10. ICMP Behavior

When a NAT sends a UDP packet towards a host on the other side of the NAT, an ICMP message may be sent in response to that packet. That ICMP message may be sent by the destination host or by any router along the network path. The NAT's default configuration SHOULD NOT filter ICMP messages based on their source IP address. Such ICMP messages SHOULD be rewritten by the NAT (specifically the IP headers and the ICMP payload) and forwarded to the appropriate internal or external host. The NAT needs to perform this function for as long as the UDP mapping is active. Receipt of any sort of ICMP message MUST NOT destroy the NAT binding. A NAT which performs the functions described in the paragraph above is referred to as "UDP Support Destination Unreachable".

There is no significant security advantage to blocking ICMP Destination Unreachable packets.

Additionally, blocking ICMP Destination Unreachable packets can interfere with application failover, UDP Path MTU Discovery (see RFC1191 [10] and RFC1435 [15]), and with traceroute. Blocking any ICMP message is discouraged, and blocking ICMP Destination Unreachable is strongly discouraged.

11. Fragmentation of Packets

When sending a packet, there are two situations that may cause IP fragmentation for packets from the inside to the outside. It is worth noting that many IP stacks do not use Path MTU Discovery with UDP packets.

11.1 Smaller Adjacent MTU

The first situation is when the MTU of the adjacent link is too small. This can occur if the NAT is doing PPPoE, or if the NAT has been configured with a small MTU to reduce serialization delay when sending large packets and small, higher-priority packets.

The packet could have its Don't Fragment bit set to 1 (DF=1) or 0 (DF=0).

If the packet has DF=1, the NAT should send back an ICMP message "fragmentation needed and DF set" message to the host as described in RFC 792 [13].

If the packet has DF=0, the NAT should fragment the packet and send the fragments, in order. This is the same function a router performs in a similar situation RFC 1812 [14].

NATs that operate as described in this section are described as "Supports Fragmentation" (abbreviated SF).

11.2 Smaller Network MTU

The second situation is when the MTU on some link in the middle of the network that is not the adjacent link is too small. If DF=0, the router adjacent to the small-MTU segment will fragment the packet and forward the fragments RFC 1812.

If DF=1, the router adjacent to the small-MTU segment will send the ICMP message "fragmentation needed and DF set" back towards the NAT. The NAT needs to forward this ICMP message to the inside host.

The classification of NATs that perform this behavior is covered in the ICMP section of this document.

12. Receiving Fragmented Packets

For a variety of reasons, a NAT may receive a fragmented UDP packet. The IP packet containing the UDP header could arrive first or last depending on network conditions, packet ordering, and the implementation of the IP stack that generated the fragments.

A NAT that is capable only of receiving UDP fragments in order (that is, with the UDP header in the first packet) and forwarding each of the fragments to the internal host is described as "Received Fragments Ordered".

A NAT that is capable of receiving UDP fragments in or out of order and forwarding the individual packets (or a reassembled packet) to the internal host is referred to as "Receive Fragments Out of Order". See the Security Considerations section of this document for a discussion of this behavior.

A NAT that is neither of these is referred to as "Receive Fragments None".

13. Requirements

The requirements in this section are aimed at minimizing the complications caused by NATs to applications such as realtime communications and online gaming.

It should be understood, however, that applications normally do not know in advance if the NAT conforms to the recommendations defined in this section. Peer-to-peer media applications still need to use normal procedures such as ICE [16] .

A NAT that supports all of the mandatory requirements of this specification (i.e., the "MUST"), is "compliant with this specification." A NAT that supports all of the requirements of this specification (i.e., included the "RECOMMENDED") is "fully compliant with all the mandatory and recommended requirements of this specification."

REQ-1 A NAT MUST have an "External NAT mapping is endpoint independent" behavior.

REQ-2 It is RECOMMENDED that a NAT have an "IP address pooling" behavior of "Paired". Note that this requirement is not applicable to NATs that do not support IP address pooling.

REQ-3 It is RECOMMENDED that a NAT have a "Port assignment" behavior of "No port preservation".
- a) NAT MAY use a "Port assignment" behavior of "Port preservation".
- b) A NAT MUST NOT have a "Port assignment" behavior of "Port overloading".
- c) If the host's source port was in the range 1-1023, it is RECOMMENDED the NAT's source port also be in the same range. If the host's source port was in the range 1024-65535, it is RECOMMENDED that the NAT's source port also be in that range.

REQ-4 It is RECOMMENDED that a NAT have a "Port parity preservation" behavior of "Yes".

REQ-5 A NAT UDP mapping timer MUST NOT expire in less than 2 minutes.
- a) The value of the NAT UDP mapping timer MAY be configurable.
- b) A default value of 5 minutes for the NAT UDP mapping timer is RECOMMENDED.

REQ-6 The NAT mapping Refresh Direction MUST have a "NAT Outbound refresh behavior" of "True".
- a) The NAT mapping Refresh Direction MAY have a "NAT Inbound refresh behavior" of "True".
- b) The NAT mapping Refresh Direction MUST have a "NAT refresh method behavior" of "Per mapping" (i.e. refresh all sessions active on a particular mapping).

REQ-7 It is RECOMMENDED that a NAT have an "External filtering is endpoint address dependent" behavior.

REQ-8 A NAT MUST support "Hairpinning".
- a) A NAT Hairpinning behavior MUST be "External source IP address and port".

REQ-9 If a NAT includes ALGs, it is RECOMMENDED that all of those ALGs be disabled by default.
- a) If a NAT includes ALGs, it is RECOMMENDED that the NAT allow the user to enable or disable each ALG separately.

REQ-10 A NAT MUST have deterministic behavior, i.e., it MUST NOT change the NAT mapping or the External External Filtering Behavior at any point in time or under any particular conditions.

REQ-11 It is RECOMMENDED that a NAT support ICMP Destination Unreachable.
  a) The ICMP timeout SHOULD be greater than 2 seconds.

REQ-12 A NAT MUST support fragmentation of packets larger than link MTU.

REQ-13 A NAT MUST support receiving in order fragments, so it MUST be "Received Fragment Ordered" or "Received Fragment Out of Order".
  a) A NAT MAY support receiving fragmented packets that are out of order and be of type "Received Fragment Out of Order".

13.1 Requirement Discussion

This section describes why each of these requirements was chosen and the consequences of violating any of them:

REQ-1 In order for UNSAF methods to work, REQ-1 needs to be met. Failure to meet REQ-1 will force the use of a Media Relay which is very often impractical.

REQ-2 This will allow applications that use multiple ports originating from the same internal IP address to also have the same external IP address. This is to avoid breaking peer-to-peer applications which are not capable of negotiating the IP address for RTP and the IP address for RTCP separately. As such it is envisioned that this requirement will become less important as applications become NAT-friendlier with time. The main reason why this requirement is here is because in a peer-to-peer application, you are subject to the other peer's mistake. In particular, in the context of SIP, if my application supports the extensions defined in RFC 3605 [9] for indicating RTP and RTCP addresses and ports separately, but the other peer does not, there may still be breakage in the form of lost of the RTP stream. This requirements will avoid the loss of RTP in this context, although the loss of RTCP may be inevitable in this particular example. It is also worth noting that RFC 3605 is unfortunately not a mandatory part of SIP (i.e., RFC 3261). This requirement will therefore address a particularly nasty problem that will prevail for a significant amount of time.

REQ-3 NATs that implement port preservation have to deal with conflicts on ports, and the multiple code paths this introduces often result in nondeterministic behavior.

c) Port preservation can work, but the NAT implementors need to be very careful that it does not become a nondeterministic NAT.

d) REQ-2b must be met in order to enable two applications on the internal side of the NAT both to use the same port to try to communicate with the same destination.

e) Certain applications expect the source UDP port to be in the well-known range. See RFC 2623 for an example.

REQ-4 This is to avoid breaking peer-to-peer applications which do not explicity and separately specify RTP and RTCP port numbers and which follow the RFC 3550 rule to decrement an odd RTP port to make it even. The same considerations as per the IP address pooling requirement apply.

REQ-5 This requirement is to ensure that the timeout is long enough to avoid too frequent timer refresh packets.

a) Configuration is desirable for adapting to specific networks and troubleshooting.

b) This default is to avoid too frequent timer refresh packets.

REQ-6 Outbound refresh is necessary for allowing the client to keep the mapping alive.

a) Inbound refresh may be useful for applications where there is no outgoing UDP traffic.

b) Using the refresh on a per mapping basis avoids the need for separate keep alive packets for all the available sessions.

REQ-7 Filtering based on the IP address is felt to have the maximum balance between security and usefulness. Filtering independently of the external IP address and port is not as secure: an unauthorized packet could get at a specific port while the port was kept open if it was lucky enough to find the port open. In theory, filtering based on both IP address and port is more secure than filtering based only on the IP address (because the external endpoint could in reality be two endpoints behind another NAT, where one of the two endpoints is an attacker). However, such a restrictive policy could interfere with certain applications that use more than one port.

REQ-8 This requirement is to allow communications between two endpoints behind the same NAT when they are trying each other's external IP addresses.

a) Using the external IP address is necessary for applications with a restrictive policy of not accepting packets from IP addresses that differ from what is expected.

REQ-9 NAT ALGs may interfere with UNSAF methods.

a) This requirement allows the user to enable ALGs which are necessary to aid operation of some applications without enabling ALGs which interfere with operation of other applications.

REQ-10 Non-deterministic NATs are very difficult to troubleshoot because they require more intensive testing. This non-deterministic behavior is the root cause of much of the uncertainty that NATs introduce about whether or not applications will work.

REQ-11 This is easy to do, is used for many things including MTU discovery and rapid detection of error conditions, and has no negative consequences.

REQ-12 Fragmented packets become more common with large video packets and should continue to work. Applications can use MTU discovery to work around this problem.

REQ-13 See Security Considerations.

14. Security Considerations

NATs are often deployed to achieve security goals. Most of the recommendations and requirements in this document do not affect the security properties of these devices, but a few of them do have security implications and are discussed in this section.

This work recommends that the timers for mapping be refreshed only on outgoing packets and does not make recommendations about whether or not inbound packets should update the timers. If inbound packets update the timers, an external attacker can keep the mapping alive forever and attack future devices that may end up with the same internal address. A device that was also the DHCP server for the private address space could mitigate this by cleaning any mappings when a DHCP lease expired. For unicast UDP traffic (the scope of this document), it may not seem relevant to support inbound timer refresh; however, for multicast UDP, the question is harder. It is expected that future documents discussing NAT behavior with multicast traffic will refine the requirements around handling of the inbound refresh timer. Some devices today do update the timers on inbound packets.

This work recommends that the NAT filters be specific to the external IP only and not the external IP and port. It can be argued that this is less secure than using the IP and port. Devices that wish to filter on IP and port do still comply with these requirements.

Non-deterministic NATs are risky from a security point of view. They are very difficult to test because they are, well, non-deterministic. Testing by a person configuring one may result in the person thinking it is behaving as desired, yet under different conditions, which an

attacker can create, the NAT may behave differently. These requirements recommend that devices be deterministic.

The work requires that NATs have an "external NAT mapping is endpoint independent" behavior. This does not reduce the security of devices. Which packets are allowed to flow across the device is determined by the external filtering behavior, which is independent of the mapping behavior.

When a fragmented packet is received from the external side and the packets are out of order so that the initial fragment does not arrive first, many systems simply discard the out of order packets. Moreover, since some networks deliver small packets ahead of large ones, there can be many out of order fragments. NATs that are capable of delivering these out of order packets are possible but they need to store the out of order fragments, which can open up a DoS opportunity. Fragmentation has been a tool used in many attacks, some involving passing fragmented packets through NATs and others involving DoS attacks based on the state needed to reassemble the fragments. NAT implementers should be aware of RFC 3128 [12] and RFC 1858 [11].

15. IANA Considerations

There are no IANA considerations.

16. IAB Considerations

The IAB has studied the problem of "Unilateral Self Address Fixing", which is the general process by which a client attempts to determine its address in another realm on the other side of a NAT through a collaborative protocol reflection mechanism [2].

This specification does not in itself constitute an UNSAF application. It consists of a series of requirements for NATs aimed at minimizing the negative impact that those devices have on peer-to-peer media applications, especially when those applications are using UNSAF methods.

Section 3 of UNSAF lists several practical issues with solutions to NAT problems. This document makes recommendations to reduce the uncertainty and problems introduced by these practical issues with NATs. In addition, UNSAF lists five architectural considerations. Although this is not an UNSAF proposal, it is interesting to consider the impact of this work on these architectural considerations.

Arch-1: The scope of this is limited to UDP packets in NATs like the ones widely deployed today. The "fix" helps constrain the variability of NATs for true UNSAF solutions such as STUN.
Arch-2: This will exit at the same rate that NATs exit. It does not imply any protocol machinery that would continue to live after NATs were gone or make it more difficult to remove them.
Arch-3: This does not reduce the overall brittleness of NATs but will hopefully reduce some of the more outrageous NAT behaviors and make it easer to discuss and predict NAT behavior in given situations.
Arch-4: This work and the results [18] of various NATs represent the most comprehensive work at IETF on what the real issues are with NATs for applications like VoIP. This work and STUN have pointed out more than anything else the brittleness NATs introduce and the difficulty of addressing these issues.
Arch-5: This work and the test results [18] provide a reference model for what any UNSAF proposal might encounter in deployed NATs.

17. Acknowledgments

The editor would like to acknowledge Bryan Ford, Pyda Srisuresh and Dan Kegel for the their draft [17] on peer-to-peer communications accross a NAT, from which a lot of the material in this specification is derived.

Dan Wing contributed substantial text on IP fragmentation and ICMP behavior.

Thanks to Rohan Mahy, Jonathan Rosenberg, Mary Barnes, Melinda Shore, Lyndsay Campbell, Geoff Huston, Jiri Kuthan, Harald Welte, Steve Casner, Robert Sanders and Spencer Dawkins for their important contributions.

18. References

18.1 Normative References

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

[2] Daigle, L. and IAB, "IAB Considerations for UNilateral Self-Address Fixing (UNSAF) Across Network Address Translation", RFC 3424, November 2002.

18.2 Informational References

[3] Srisuresh, P. and M. Holdrege, "IP Network Address Translator

(NAT) Terminology and Considerations", RFC 2663, August 1999.

[4] Srisuresh, P. and K. Egevang, "Traditional IP Network Address Translator (Traditional NAT)", RFC 3022, January 2001.

[5] Holdrege, M. and P. Srisuresh, "Protocol Complications with the IP Network Address Translator", RFC 3027, January 2001.

[6] 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.

[7] Rosenberg, J., Weinberger, J., Huitema, C. and R. Mahy, "STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)", RFC 3489, March 2003.

[8] Schulzrinne, H., Casner, S., Frederick, R. and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", RFC 3550, July 2003.

[9] Huitema, C., "Real Time Control Protocol (RTCP) attribute in Session Description Protocol (SDP)", RFC 3605, October 2003.

[10] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191, November 1990.

[11] Ziemba, G., Reed, D. and P. Traina, "Security Considerations for IP Fragment Filtering", RFC 1858, October 1995.

[12] Miller, I., "Protection Against a Variant of the Tiny Fragment Attack (RFC 1858)", RFC 3128, June 2001.

[13] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, September 1981.

[14] Baker, F., "Requirements for IP Version 4 Routers", RFC 1812, June 1995.

[15] Knowles, S., "IESG Advice from Experience with Path MTU Discovery", March 1993.

[16] Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Methodology for Network Address Translator (NAT) Traversal for the Session Initiation Protocol (SIP)", draft-ietf-mmusic-ice-03 (work in progress), October 2004.

[17] Ford, B., Srisuresh, P. and D. Kegel, "State of Peer-to-Peer(P2P) communication across Network Address

Translators(NATs)", draft-srisuresh-behave-p2p-state-00 (work in progress), December 2004.

[18] Jennings, C., "NAT Classification Results using STUN", draft-jennings-midcom-stun-results-02 (work in progress), October 2004.

[19] "Packet-based Multimedia Communications Systems", ITU-T Recommendation H.323, July 2003.

[20] Ford, B. and D. Andersen, "Nat Check Web Site: http://midcom-p2p.sourceforge.net", June 2004.

Authors' Addresses

Francois Audet (editor)
Nortel Networks
4655 Great America Parkway
Santa Clara, CA 95054
US

Phone: +1 408 495 3756
EMail: audet@nortel.com

Cullen Jennings
Cisco Systems
170 West Tasman Drive
MS: SJC-21/2
San Jose, CA 95134
US

Phone: +1 408 902 3341
EMail: fluffy@cisco.com

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2005). This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.