          Network Address Translation - Protocol Translation (NAT-PT)

Status of this Memo

Copyright Notice

Abstract

   This document specifies an IPv4-to-IPv6 transition mechanism, in
   addition to those already specified in [TRANS]. This solution
   attempts to provide transparent routing, as defined in [NAT-TERM], to
   end-nodes in V6 realm trying to communicate with end-nodes in V4
   realm and vice versa. This is achieved using a combination of Network
   Address Translation and Protocol Translation. The scheme described
   does not mandate dual-stacks (i.e., IPv4 as well as V6 protocol
   support) or special purpose routing requirements (such as requiring
   tunneling support) on end nodes. This scheme is based on a
   combination of address translation theme as described in [NAT-TERM]
   and V6/V4 protocol translation theme as described in [SIIT].

Acknowledgements

Table of Contents

1. Introduction

   IPv6 is a new version of the IP protocol designed to modernize IPv4
   which was designed in the 1970s. IPv6 has a number of advantages over
   IPv4 that will allow for future Internet growth and will simplify IP
   configuration and administration. IPv6 has a larger address space
   than IPv4, an addressing model that promotes aggressive route
   aggregation and a powerful autoconfiguration mechanism. In time, it
   is expected that Internet growth and a need for a plug-and-play
   solution will result in widespread adoption of IPv6.

There is expected to be a long transition period during which it will
be necessary for IPv4 and IPv6 nodes to coexist and communicate. A
strong, flexible set of IPv4-to-IPv6 transition and coexistence
mechanisms will be required during this transition period.

The SIIT proposal [SIIT] describes a protocol translation mechanism
that allows communication between IPv6-only and IPv4-only nodes via
protocol independent translation of IPv4 and IPv6 datagrams,
requiring no state information for the session. The SIIT proposal
assumes that V6 nodes are assigned a V4 address for communicating
with V4 nodes, and does not specify a mechanism for the assignment of
these addresses.

NAT-PT uses a pool of V4 addresses for assignment to V6 nodes on a
dynamic basis as sessions are initiated across V4-V6 boundaries. The
V4 addresses are assumed to be globally unique. NAT-PT with private
V4 addresses is outside the scope of this document and for further
study. NAT-PT binds addresses in V6 network with addresses in V4
network and vice versa to provide transparent routing [NAT-TERM] for
the datagrams traversing between address realms. This requires no
changes to end nodes and IP packet routing is completely transparent
[NAT-TERM] to end nodes. It does, however, require NAT-PT to track
the sessions it supports and mandates that inbound and outbound
datagrams pertaining to a session traverse the same NAT-PT router.
You will note that the topology restrictions on NAT-PT are the same
with those described for V4 NATs in [NAT-TERM]. Protocol translation
details specified in [SIIT] would be used to extend address
translation with protocol syntax/semantics translation. A detailed
applicability statement for NAT-PT may be found at the end of this
document in section 7.

By combining SIIT protocol translation with the dynamic address
translation capabilities of NAT and appropriate ALGs, NAT-PT provides
a complete solution that would allow a large number of commonly used
applications to interoperate between IPv6-only nodes and IPv4-only

A fundamental assumption for NAT-PT is only to be use when no other
native IPv6 or IPv6 over IPv4 tunneled means of communication is
possible. In other words the aim is to only use translation between
IPv6 only nodes and IPv4 only nodes, while translation between IPv6
only nodes and the IPv4 part of a dual stack node should be avoided
over other alternatives.

2. Terminology

The majority of terms used in this document are borrowed almost as is
from [NAT-TERM]. The following lists terms specific to this document.

2.1 Network Address Translation (NAT)

   The term NAT in this document is very similar to the IPv4 NAT
   described in [NAT-TERM], but is not identical. IPv4 NAT translates
   one IPv4 address into another IPv4 address. In this document, NAT
   refers to translation of an IPv4 address into an IPv6 address and
   vice versa.

   While the V4 NAT [NAT-TERM] provides routing between private V4 and
   external V4 address realms, NAT in this document provides routing
   between a V6 address realm and an external V4 address realm.

2.2 NAT-PT flavors

   Just as there are various flavors identified with V4 NAT in [NAT-
   TERM], the following NAT-PT variations may be identified in this
   document.

2.2.1 Traditional NAT-PT

   Traditional-NAT-PT would allow hosts within a V6 network to access
   hosts in the V4 network. In a traditional-NAT-PT, sessions are uni-
   directional, outbound from the V6 network. This is in contrast with
   Bi-directional-NAT-PT, which permits sessions in both inbound and
   outbound directions.

   Just as with V4 traditional-NAT, there are two variations to
   traditional-NAT-PT, namely Basic-NAT-PT and NAPT-PT.

   With Basic-NAT-PT, a block of V4 addresses are set aside for
   translating addresses of V6 hosts as they originate sessions to the
   V4 hosts in external domain. For packets outbound from the V6 domain,
   the source IP address and related fields such as IP, TCP, UDP and
   ICMP header checksums are translated. For inbound packets, the
   destination IP address and the checksums as listed above are
   translated.

   NAPT-PT extends the notion of translation one step further by also
   translating transport identifier (e.g., TCP and UDP port numbers,
   ICMP query identifiers). This allows the transport identifiers of a
   number of V6 hosts to be multiplexed into the transport identifiers
   of a single assigned V4 address. NAPT-PT allows a set of V6 hosts to
   share a single V4 address. Note that NAPT-PT can be combined with
   Basic-NAT-PT so that a pool of external addresses are used in
   conjunction with port translation.

For packets outbound from the V6 network, NAPT-PT would translate the
source IP address, source transport identifier and related fields
such as IP, TCP, UDP and ICMP header checksums. Transport identifier
can be one of TCP/UDP port or ICMP query ID. For inbound packets, the
destination IP address, destination transport identifier and the IP
and transport header checksums are translated.

2.2.2 Bi-Directional-NAT-PT

With Bi-directional-NAT-PT, sessions can be initiated from hosts in
V4 network as well as the V6 network. V6 network addresses are bound
to V4 addresses, statically or dynamically as connections are
established in either direction. The name space (i.e., their Fully
Qualified Domain Names) between hosts in V4 and V6 networks is
assumed to be end-to-end unique. Hosts in V4 realm access V6-realm
hosts by using DNS for address resolution. A DNS-ALG [DNS-ALG] must
be employed in conjunction with Bi-Directional-NAT-PT to facilitate
name to address mapping. Specifically, the DNS-ALG must be capable
of translating V6 addresses in DNS Queries and responses into their
V4-address bindings, and vice versa, as DNS packets traverse between
V6 and V4 realms.

2.3 Protocol Translation (PT)

PT in this document refers to the translation of an IPv4 packet into
a semantically equivalent IPv6 packet and vice versa. Protocol
translation details are described in [SIIT].

2.4 Application Level Gateway (ALG)

Application Level Gateway (ALG) [NAT-TERM] is an application specific
agent that allows a V6 node to communicate with a V4 node and vice
versa. Some applications carry network addresses in payloads. NAT-PT
is application unaware and does not snoop the payload. ALG could work
in conjunction with NAT-PT to provide support for many such
applications.

2.5 Requirements

The keywords MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD,
SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL, when they appear in this
document, are to be interpreted as described in [KEYWORDS].

3. Traditional-NAT-PT Operation (V6 to V4)

   NAT-PT offers a straight forward solution based on transparent
   routing [NAT-TERM] and address/protocol translation, allowing a large
   number of applications in V6 and V4 realms to inter-operate without
   requiring any changes to these applications.

   In the following paragraphs we describe the operation of
   traditional-NAT-PT and the way that connections can be initiated from
   a host in IPv6 domain to a host in IPv4 domain through a
   traditional-NAT-PT

3.1 Basic-NAT-PT Operation

```
        [IPv6-B]-+
                 |                      +==============+
        [IPv6-A]-+-[NAT-PT]---------| IPv4 network |--[IPv4-C]
                 |                      +==============+
              (pool of v4 addresses)


                 Figure 1: IPv6 to IPv4 communication
         Node IPv6-A has an IPv6 address -> FEDC:BA98::7654:3210
         Node IPv6-B has an IPv6 address -> FEDC:BA98::7654:3211
            Node IPv4-C has an IPv4 address -> 132.146.243.30
```
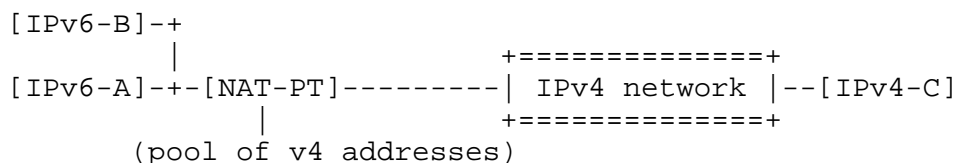
   NAT-PT has a pool of addresses including the IPv4 subnet
   120.130.26/24

   The V4 addresses in the address pool could be allocated one-to-one to
   the V6 addresses of the V6 end nodes in which case one needs as many
   V4 addresses as V6 end points. In this document we assume that the V6
   network has less V4 addresses than V6 end nodes and thus dynamic
   address allocation is required for at least some of them.

   Say the IPv6 Node A wants to communicate with the IPv4 Node C. Node
   A creates a packet with:

      Source Address, SA=FEDC:BA98::7654:3210 and Destination
      Address, DA = PREFIX::132.146.243.30

   NOTE: The prefix PREFIX::/96 is advertised in the stub domain by the
   NAT-PT, and packets addressed to this PREFIX will be routed to the
   NAT-PT. The pre-configured PREFIX only needs to be routable within
   the IPv6 stub domain and as such it can be any routable prefix that
   the network administrator chooses.

   The packet is routed via the NAT-PT gateway, where it is translated
   to IPv4.

If the outgoing packet is not a session initialisation packet, the
NAT-PT SHOULD already have stored some state about the related
session, including assigned IPv4 address and other parameters for the
translation. If this state does not exist, the packet SHOULD be
silently discarded.

If the packet is a session initialisation packet, the NAT-PT locally
allocates an address (e.g: 120.130.26.10) from its pool of
addresses and the packet is translated to IPv4. The translation
parameters are cached for the duration of the session and the IPv6 to
IPv4 mapping is retained by NAT-PT.

The resulting IPv4 packet has SA=120.130.26.10 and DA=132.146.243.30.
Any returning traffic will be recognised as belonging to the same
session by NAT-PT. NAT-PT will use the state information to translate
the packet, and the resulting addresses will be
SA=PREFIX::132.146.243.30, DA=FEDC:BA98::7654:3210. Note that this
packet can now be routed inside the IPv6-only stub network as normal.

3.2 NAPT-PT Operation

NAPT-PT, which stands for "Network Address Port Translation +
Protocol Translation", would allow V6 nodes to communicate with the
V4 nodes transparently using a single V4 address. The TCP/UDP ports
of the V6 nodes are translated into TCP/UDP ports of the registered
V4 address.

While NAT-PT support is limited to TCP, UDP and other port
multiplexing type of applications, NAPT-PT solves a problem that is
inherent with NAT-PT. That is, NAT-PT would fall flat when the pool
of V4 addresses assigned for translation purposes is exhausted. Once
the address pool is exhausted, newer V6 nodes cannot establish
sessions with the outside world anymore. NAPT-PT, on the other hand,
will allow for a maximum of 63K TCP and 63K UDP sessions per IPv4
address before having no TCP and UDP ports left to assign.

To modify the example sited in figure 1, we could have NAPT-PT on the
border router (instead of NAT-PT) and all V6 addresses could be
mapped to a single v4 address 120.130.26.10.

IPv6 Node A would establish a TCP session with the IPv4 Node C as
follows:

Node A creates a packet with:

Source Address, SA=FEDC:BA98::7654:3210 , source TCP port = 3017 and
Destination Address, DA = PREFIX::132.146.243.30, destination TCP
port = 23.

When the packet reaches the NAPT-PT box, NAPT-PT would assign one of
the TCP ports from the assigned V4 address to translate the tuple of
(Source Address, Source TCP port) as follows:

    SA=120.130.26.10, source TCP port = 1025 and
    DA=132.146.243.30, destination TCP port = 23.

The returning traffic from 132.146.243.30, TCP port 23 will be
recognised as belonging to the same session and will be translated
back to V6 as follows:

    SA = PREFIX::132.146.243.30, source TCP port = 23;
    DA = FEDC:BA98::7654:3210 , destination TCP port = 3017

Inbound NAPT-PT sessions are restricted to one server per service,
assigned via static TCP/UDP port mapping. For example, the Node
[IPv6-A] in figure 1 may be the only HTTP server (port 80) in the V6
domain. Node [IPv4-C] sends a packet:

    SA=132.146.243.30, source TCP port = 1025 and
    DA=120.130.26.10, destination TCP port = 80

NAPT-PT will translate this packet to:

    SA=PREFIX::132.146.243.30, source TCP port = 1025
    DA=FEDC:BA98::7654:3210, destination TCP port = 80

In the above example, note that all sessions which reach NAPT-PT with
a destination port of 80 will be redirected to the same node [IPv6-
A].

4. Use of DNS-ALG for Address Assignment

An IPv4 address is assigned by NAT-PT to a V6 node when NAT-PT
identifies the start of session, inbound or outbound. Identification
of the start of a new inbound session is performed differently than
for outbound sessions. However, the same V4 address pool is used for
assignment to V6 nodes, irrespective of whether a session is
initiated outbound from a V6 node or initiated inbound from a V4
node.

Policies determining what type of sessions are allowed and in which
direction and from/to which nodes is out of the scope of this
document.

IPv4 name to address mappings are held in the DNS with "A" records.
IPv6 name to address mappings are at the moment held in the DNS with
"AAAA" records. "A6" records have also been defined but at the time
of writing they are neither fully standardized nor deployed.

In any case, the DNS-ALG's principle of operation described in this
section is the same with either "AAAA" or "A6" records. The only
difference is that a name resolution using "A6" records may require
more than one query - reply pairs. The DNS-ALG SHOULD, in that case,
track all the replies in the transaction before translating an "A6"
record to an "A" record.

One of the aims of NAT-PT design is to only use translation when
there is no other means of communication, such as native IPv6 or some
form of tunneling. For the following discussion NAT-PT, in addition
to the IPv4 connectivity that it has it may also have a native IPv6
and/or a tunneled IPv6 connection.

4.1 V4 Address assignment for incoming connections (V4 to V6)

```
    [DNS]--+
           |                    [DNS]------[DNS]-------[DNS]
    [IPv6-B]-+                             |          |
           |                 +==============+         |
    [IPv6-A]-+----[NAT-PT]------| IPv4 network |--[IPv4-C]
              |                 +==============+
          (pool of v4 addresses)


              Figure 2: IPv4 to IPv6 communication
       Node IPv6-A has an IPv6 address -> FEDC:BA98::7654:3210
       Node IPv6-B has an IPv6 address -> FEDC:BA98::7654:3211
          Node IPv4-C has an IPv4 address -> 132.146.243.30
```
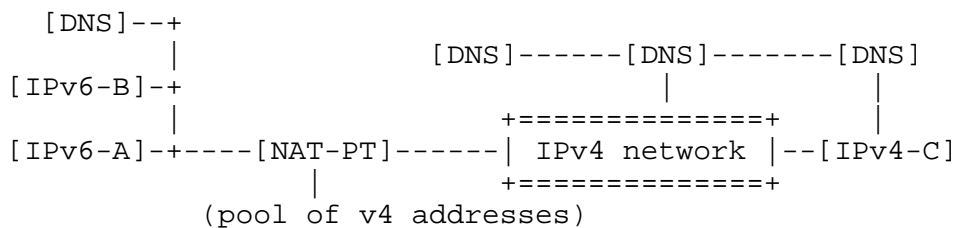
NAT-PT has a pool of addresses including the IPv4 subnet
120.130.26/24

In figure 2 above, when Node C's name resolver sends a name look up
request for Node A, the lookup query is directed to the DNS server on
the V6 network. Considering that NAT-PT is residing on the border
router between V4 and V6 networks, this request datagram would
traverse through the NAT-PT router. The DNS-ALG on the NAT-PT device
would modify DNS Queries for A records going into the V6 domain as
follows: (Note that a TCP/UDP DNS packet is recognised by the fact
that its source or destination port number is 53)

   a) For Node Name to Node Address Query requests: Change the Query
      type from "A" to "AAAA" or "A6".

   b) For Node address to Node name query requests: Replace the
      string "IN-ADDR.ARPA" with the string "IP6.INT". Replace the
      V4 address octets (in reverse order) preceding the string "IN-
      ADDR.ARPA" with the corresponding V6 address (if there exists a
      map) octets in reverse order.

   In the opposite direction, when a DNS response traverses from the DNS
   server on the V6 network to the V4 node, the DNS-ALG once again
   intercepts the DNS packet and would:

   a) Translate DNS responses for "AAAA" or "A6" records into "A"
      records, (only translate "A6" records when the name has
      completely been resolved)
   b) Replace the V6 address resolved by the V6 DNS with the V4
      address internally assigned by the NAT-PT router.

   If a V4 address is not previously assigned to this V6 node, NAT-PT
   would assign one at this time. As an example say IPv4-C attempts to
   initialise a session with node IPv6-A by making a name lookup ("A"
   record) for Node-A . The name query goes to the local DNS and from
   there it is propagated to the DNS server of the IPv6 network. The
   DNS-ALG intercepts and translates the "A" query to "AAAA" or "A6"
   query and then forwards it to the DNS server in the IPv6 network
   which replies as follows: (The example uses AAAA records for
   convenience)

      Node-A   AAAA   FEDC:BA98::7654:3210,

   this is returned by the DNS server and gets intercepted and
   translated by the DNS-ALG to:

      Node-A   A   120.130.26.1

   The DNS-ALG also holds the mapping between FEDC:BA98::7654:3210 and
   120.130.26.1 in NAT-PT. The "A" record is then returned to Node-C.
   Node-C can now initiate a session as follows:

      SA=132.146.243.30, source TCP port = 1025 and
      DA=120.130.26.1, destination TCP port = 80

   the packet will be routed to NAT-PT, which since it already holds a
   mapping between FEDC:BA98::7654:3210 and 120.130.26.1 can translate
   the packet to:

      SA=PREFIX::132.146.243.30, source TCP port = 1025
      DA=FEDC:BA98::7654:3210, destination TCP port = 80

   the communication can now proceed as normal.

The TTL values on all DNS resource records (RRs) passing through
NAT-PT SHOULD be set to 0 so that DNS servers/clients do not cache
temporarily assigned RRs. Note, however, that due to some buggy DNS
client implementations a value of 1 might in some cases work better.
The TTL values should be left unchanged for statically mapped
addresses.

Address mappings for incoming sessions, as described above, are
subject to denial of service attacks since one can make multiple
queries for nodes residing in the V6 network causing the DNS-ALG to
map all V4 addresses in NAT-PT and thus block legitimate incoming
sessions. Thus, address mappings for incoming sessions should time
out to minimise the effect of denial of service attacks.
Additionally, one IPv4 address (using NAPT-PT, see 3.2) could be
reserved for outgoing sessions only to minimise the effect of such
attacks to outgoing sessions.

4.2 V4 Address assignment for outgoing connections (V6 to V4)

V6 nodes learn the address of V4 nodes from the DNS server in the V4
domain or from the DNS server internal to the V6 network. We
recommend that DNS servers internal to V6 domains maintain a mapping
of names to IPv6 addresses for internal nodes and possibly cache
mappings for some external nodes. In the case where the DNS server in
the v6 domain contains the mapping for external V4 nodes, the DNS
queries will not cross the V6 domain and that would obviate the need
for DNS-ALG intervention. Otherwise, the queries will cross the V6
domain and are subject to DNS-ALG intervention. We recommend
external DNS servers in the V4 domain cache name mapping for external
nodes (i.e., V4 nodes) only. Zone transfers across IPv4 - IPv6
boundaries are strongly discouraged.

In the case of NAPT-PT, a TCP/UDP source port is assigned from the
registered V4 address upon detection of each new outbound session.

We saw that a V6 node that needs to communicate with a V4 node needs
to use a specific prefix (PREFIX::/96) in front of the IPv4 address
of the V4 node. The above technique allows the use of this PREFIX
without any configuration in the nodes.

To create another example from Figure 2 say Node-A wants to set up a
session with Node-C. For this Node-A starts by making a name look-up
("AAAA" or "A6" record) for Node-C.

Since Node-C may have IPv6 and/or IPv4 addresses, the DNS-ALG on the
NAT-PT device forwards the original AAAA/A6 query to the external DNS
system unchanged, as well as an A query for the same node. If an
AAAA/A6 record exists for the destination, this will be returned to

NAT-PT which will forward it, also unchanged, to the originating
host.

If there is an A record for Node-C the reply also returns to the
NAT-PT. The DNS-ALG then, translates the reply adding the appropriate
PREFIX and forwards it to the originating device with any IPv6
addresses that might have learned. So, if the reply is

```
    NodeC    A      132.146.243.30, it is translated to
    NodeC    AAAA   PREFIX::132.146.243.30 or to
    NodeC    A6     PREFIX::132.146.243.30
```

Now Node A can use this address like any other IPv6 address and the
V6 DNS server can even cache it as long as the PREFIX does not
change.

An issue here is how the V6 DNS server in the V6 stub domain talks to
the V4 domain outside the V6 stub domain. Remember that there are no
dual stack nodes here. The external V4 DNS server needs to point to a
V4 address, part of the V4 pool of addresses, available to NAT-PT.
NAT-PT keeps a one-to-one mapping between this V4 address and the V6
address of the internal V6 DNS server. In the other direction, the V6
DNS server points to a V6 address formed by the IPv4 address of the
external V4 DNS servers and the prefix (PREFIX::/96) that indicates
non IPv6 nodes. This mechanism can easily be extended to accommodate
secondary DNS servers.

Note that the scheme described in this section impacts DNSSEC. See
section 7.5 of this document for details.

5. Protocol Translation Details

The IPv4 and ICMPv4 headers are similar to their V6 counterparts but
a number of field are either missing, have different meaning or
different length. NAT-PT SHOULD translate all IP/ICMP headers from v4
to v6 and vice versa in order to make end-to-end IPv6 to IPv4
communication possible. Due to the address translation function and
possible port multiplexing, NAT-PT SHOULD also make appropriate
adjustments to the upper layer protocol (TCP/UDP) headers. A separate
section on FTP-ALG describes the changes FTP-ALG would make to FTP
payload as an FTP packet traverses from V4 to V6 realm or vice versa.

Protocol Translation details are described in [SIIT], but there are
some modifications required to SIIT because of the fact that NAT-PT
also performs Network Address Translation.

5.1 Translating IPv4 headers to IPv6 headers

   This is done exactly the same as in SIIT apart from the following
   fields:

      Source Address:
         The low-order 32 bits is the IPv4 source address. The high-
         order 96 bits is the designated PREFIX for all v4
         communications. Addresses using this PREFIX will be routed
         to the NAT-PT gateway (PREFIX::/96)

      Destination Address:
         NAT-PT retains a mapping between the IPv4 destination
         address and the IPv6 address of the destination node. The
         IPv4 destination address is replaced by the IPv6 address
         retained in that mapping.

5.2 Translating IPv6 headers to IPv4 headers

   This is done exactly the same as in SIIT apart from the Source
   Address which should be determined as follows:

      Source Address:
         The NAT-PT retains a mapping between the IPv6 source address
         and an IPv4 address from the pool of IPv4 addresses
         available. The IPv6 source address is replaced by the IPv4
         address retained in that mapping.

      Destination Address:
         IPv6 packets that are translated have a destination address
         of the form PREFIX::IPv4/96. Thus the low-order 32 bits of
         the IPv6 destination address is copied to the IPv4
         destination address.

5.3 TCP/UDP/ICMP Checksum Update

   NAT-PT retains mapping between IPv6 address and an IPv4 address from
   the pool of IPv4 addresses available. This mapping is used in the
   translation of packets that go through NAT-PT.

   The following sub-sections describe TCP/UDP/ICMP checksum update
   procedure in NAT-PT, as packets are translated from V4 to V6 and vice
   versa.

5.3.1 TCP/UDP/ICMP Checksum Update from IPv4 to IPv6

   UDP checksums, when set to a non-zero value, and TCP checksum SHOULD
   be recalculated to reflect the address change from v4 to v6. The
   incremental checksum adjustment algorithm may be borrowed from [NAT].
   In the case of NAPT-PT, TCP/UDP checksum should be adjusted to
   account for the address and TCP/UDP port changes, going from V4 to V6
   address.

   When the checksum of a V4 UDP packet is set to zero, NAT-PT MUST
   evaluate the checksum in its entirety for the V6-translated UDP
   packet. If a V4 UDP packet with a checksum of zero arrives in
   fragments, NAT-PT MUST await all the fragments until they can be
   assembled into a single non-fragmented packet and evaluate the
   checksum prior to forwarding the translated V6 UDP packet.

   ICMPv6, unlike ICMPv4, uses a pseudo-header, just like UDP and TCP
   during checksum computation. As a result, when the ICMPv6 header
   checksum is computed [SIIT], the checksum needs to be adjusted to
   account for the additional pseudo-header. Note, there may also be
   adjustments required to the checksum due to changes in the source and
   destination addresses (and changes in TCP/UDP/ICMP identifiers in the
   case of NAPT-PT) of the payload carried within ICMP.

5.3.2 TCP/UDP/ICMP Checksum Update from IPv6 to IPv4

   TCP and UDP checksums SHOULD be recalculated to reflect the address
   change from v6 to v4. The incremental checksum adjustment algorithm
   may be borrowed from [NAT]. In the case of NAPT-PT, TCP/UDP checksums
   should be adjusted to account for the address and TCP/UDP port
   changes, going from V6 to V4 addresses. For UDP packets, optionally,
   the checksum may simply be changed to zero.

   The checksum calculation for a V4 ICMP header needs to be derived
   from the V6 ICMP header by running the checksum adjustment algorithm
   [NAT] to remove the V6 pseudo header from the computation. Note, the
   adjustment must additionally take into account changes to the
   checksum as a result of updates to the source and destination
   addresses (and transport ports in the case of NAPT-PT) made to the
   payload carried within ICMP.

6. FTP Application Level Gateway (FTP-ALG) Support

   Because an FTP control session carries, in its payload, the IP
   address and TCP port information for the data session, an FTP-ALG is
   required to provide application level transparency for this popular
   Internet application.

In the FTP application running on a legacy V4 node, arguments to the
FTP PORT command and arguments in PASV response(successful) include
an IP V4 address and a TCP port, both represented in ASCII as
h1,h2,h3,h4,p1,p2. However, [FTP-IPV6] suggests EPRT and EPSV command
extensions to FTP, with an intent to eventually retire the use of
PORT and PASV commands. These extensions may be used on a V4 or V6
node. FTP-ALG, facilitating transparent FTP between V4 and V6 nodes,
works as follows.

6.1 Payload modifications for V4 originated FTP sessions

A V4 host may or may not have the EPRT and EPSV command extensions
implemented in its FTP application. If a V4 host originates the FTP
session and uses PORT or PASV command, the FTP-ALG will translate
these commands into EPRT and EPSV commands respectively prior to
forwarding to the V6 node. Likewise, EPSV response from V6 nodes will
be translated into PASV response prior to forwarding to V4 nodes.
The format of EPRT and EPSV commands and EPSV response may be
specified as follows[FTP-IPV6].

    EPRT<space><d><net-prt><d><net-addr><d><tcp-port><d>
    EPSV<space><net-prt>
          (or)
    EPSV<space>ALL

    Format of EPSV response(Positive): 229 <text indicating
    extended passive mode> (<d><d><d><tcp-port><d>)

PORT command from a V4 node is translated into EPRT command, by
setting the protocol <net-prt> field to AF #2 (IPV6) and translating
the V4 host Address (represented as h1,h2,h3,h4) into its NAT-PT
assigned V6 address in string notation, as defined in [V6ADDR] in the
<net-addr> field. TCP port represented by p1,p2 in PORT command must
be specified as a decimal <tcp-port> in the EPRT command. Further,
<tcp-port> translation may also be required in the case of NAPT-PT.
PASV command from a V4 node is be translated into a EPSV command with
the <net-prt> argument set to AF #2. EPSV response from a V6 node is
translated into PASV response prior to forwarding to the target V4
host.

If a V4 host originated the FTP session and was using EPRT and EPSV
commands, the FTP-ALG will simply translate the parameters to these
commands, without altering the commands themselves. The protocol
Number <net-prt> field will be translated from AF #1 to AF #2.
<net-addr> will be translated from the V4 address in ASCII to its
NAT-PT assigned V6 address in string notation as defined in [V6ADDR].
<tcp-port> argument in EPSV response requires translation only in the
case of NAPT-PT.

6.2 Payload modifications for V6 originated FTP sessions

   If a V6 host originates the FTP session, however, the FTP-ALG has two
   approaches to pursue. In the first approach, the FTP-ALG will leave
   the command strings "EPRT" and "EPSV" unaltered and simply translate
   the <net-prt>, <net-addr> and <tcp-port> arguments from V6 to its
   NAT-PT (or NAPT-PT) assigned V4 information. <tcp-port> is translated
   only in the case of NAPT-PT. Same goes for EPSV response from V4
   node. This is the approach we recommend to ensure forward support for
   RFC 2428. However, with this approach, the V4 hosts are mandated to
   have their FTP application upgraded to support EPRT and EPSV
   extensions to allow access to V4 and V6 hosts, alike.

   In the second approach, the FTP-ALG will translate the command
   strings "EPRT" and "EPSV" and their parameters from the V6 node into
   their equivalent NAT-PT assigned V4 node info and attach to "PORT"
   and "PASV" commands prior to forwarding to V4 node. Likewise, PASV
   response from V4 nodes is translated into EPSV response prior to
   forwarding to the target V6 nodes. However, the FTP-ALG would be
   unable to translate the command "EPSV<space>ALL" issued by V6 nodes.
   In such a case, the V4 host, which receives the command, may return
   an error code indicating unsupported function. This error response
   may cause many RFC 2428 compliant FTP applications to simply fail,
   because EPSV support is mandated by RFC 2428. The benefit of this
   approach, however, is that is does not impose any FTP upgrade
   requirements on V4 hosts.

6.3 Header updates for FTP control packets

   All the payload translations considered in the previous sections are
   based on ASCII encoded data. As a result, these translations may
   result in a change in the size of packet.

   If the new size is the same as the previous, only the TCP checksum
   needs adjustment as a result of the payload translation. If the new
   size is different from the previous, TCP sequence numbers should also
   be changed to reflect the change in the length of the FTP control
   session payload. The IP packet length field in the V4 header or the
   IP payload length field in the V6 header should also be changed to
   reflect the new payload size. A table is used by the FTP-ALG to
   correct the TCP sequence and acknowledgement numbers in the TCP
   header for control packets in both directions.

   The table entries should have the source address, source data port,
   destination address and destination data port for V4 and V6 portions
   of the session, sequence number delta for outbound control packets
   and sequence number delta for inbound control packets.

   The sequence number for an outbound control packet is increased by
   the outbound sequence number delta, and the acknowledgement number
   for the same outbound packet is decreased by the inbound sequence
   number delta. Likewise, the sequence number for an inbound packet is
   increased by the inbound sequence number delta and the
   acknowledgement number for the same inbound packet is decreased by
   the outbound sequence number delta.

7. NAT-PT Limitations and Future Work

   All limitations associated to NAT [NAT-TERM] are also associated to
   NAT-PT. Here are the most important of them in detail, as well as
   some unique to NAT-PT.

7.1 Topology limitations

   There are limitations to using the NAT-PT translation method. It is
   mandatory that all requests and responses pertaining to a session be
   routed via the same NAT-PT router. One way to guarantee this would be
   to have NAT-PT based on a border router that is unique to a stub
   domain, where all IP packets are either originated from the domain or
   destined to the domain. This is a generic problem with NAT and it is
   fully described in [NAT-TERM].

   Note, this limitation does not apply to packets originating from or
   directed to dual-stack nodes that do not require packet translation.
   This is because in a dual-stack set-up, IPv4 addresses implied in a
   V6 address can be identified from the address format PREFIX::x.y.z.w
   and a dual-stack router can accordingly route a packet between v4 and
   dual-stack nodes without tracking state information.

   This should also not affect IPv6 to IPv6 communication and in fact
   only actually use translation when no other means of communication is
   possible. For example NAT-PT may also have a native IPv6 connection
   and/or some kind of tunneled IPv6 connection. Both of the above
   connections should be preferred over translation when possible. The
   above makes sure that NAT-PT is a tool only to be used to assist
   transition to native IPv6 to IPv6 communication.

7.2 Protocol Translation Limitations

   A number of IPv4 fields have changed meaning in IPv6 and translation
   is not straightforward. For example, the option headers semantics and
   syntax have changed significantly in IPv6. Details of IPv4 to IPv6
   Protocol Translation can be found in [SIIT].

7.3 Impact of Address Translation

   Since NAT-PT performs address translation, applications that carry
   the IP address in the higher layers will not work. In this case
   Application Layer Gateways (ALG) need to be incorporated to provide
   support for those applications. This is a generic problem with NAT
   and it is fully described in [NAT-TERM].

7.4 Lack of end-to-end security

   One of the most important limitations of the NAT-PT proposal is the
   fact that end-to-end network layer security is not possible. Also
   transport and application layer security may not be possible for
   applications that carry IP addresses to the application layer. This
   is an inherent limitation of the Network Address Translation
   function.

   Independent of NAT-PT, end-to-end IPSec security is not possible
   across different address realms. The two end-nodes that seek IPSec
   network level security must both support one of IPv4 or IPv6.

7.5 DNS Translation and DNSSEC

   The scheme described in section 4.2 involves translation of DNS
   messages. It is clear that this scheme can not be deployed in
   combination with secure DNS. I.e., an authoritative DNS name server
   in the V6 domain cannot sign replies to queries that originate from
   the V4 world. As a result, an V4 end-node that demands DNS replies
   to be signed will reject replies that have been tampered with by
   NAT-PT.

   The good news, however, is that only servers in V6 domain that need
   to be accessible from the V4 world pay the price for the above
   limitation, as V4 end-nodes may not access V6 servers due to DNS
   replies not being signed.

   Also note that zone transfers between DNS-SEC servers within the same
   V6 network are not impacted.

   Clearly, with DNS SEC deployment in DNS servers and end-host
   resolvers, the scheme suggested in this document would not work.

8. Applicability Statement

   NAT-PT can be a valuable transition tool at the border of a stub
   network that has been deployed as an IPv6 only network when it is
   connected to an Internet that is either V4-only or a combination of
   V4 and V6.

NAT-PT, in its simplest form, without the support of DNS-ALG,
provides one way connectivity between an IPv6 stub domain and the
IPv4 world meaning that only sessions initialised by IPv6 nodes
internal to the IPv6 stub domain can be translated, while sessions
initiated by IPv4 nodes are dropped. This makes NAT-PT a useful
tool to IPv6 only stub networks that need to be able to maintain
connectivity with the IPv4 world without the need to deploy servers
visible to the IPv4 world.

NAT-PT combined with a DNS-ALG provides bi-directional connectivity
between the IPv6 stub domain and the IPv4 world allowing sessions to
be initialised by IPv4 nodes outside the IPv6 stub domain. This
makes NAT-PT useful for IPv6 only stub networks that need to deploy
servers visible to the IPv4 world.

Some applications count on a certain degree of address stability for
their operation. Dynamic address reuse by NAT-PT might not be
agreeable for these applications. For hosts running such address
critical applications, NAT-PT may be configured to provide static
address mapping between the host's V6 address and a specific V4
address. This will ensure that address related changes by NAT-PT do
not become a significant source of operational failure.

9. Security Considerations

   Section 7.4 of this document states that end-to-end network and
   transport layer security are not possible when a session is
   intercepted by a NAT-PT. Also application layer security may not be
   possible for applications that carry IP addresses in the application
   layer.

   Section 7.5 of this document states that the DNS-ALG can not be
   deployed in combination with secure DNS.

   Finally, all of the security considerations described in [NAT-TERM]
   are applicable to this document as well.

10. REFERENCES

   [DNS-ALG]   Srisuresh, P., Tsirtsis, G., Akkiraju, P. and A.
               Heffernan, "DNS extensions to Network Address Translators
               (DNS_ALG)", RFC 2694, September 1999.

   [DNSSEC]   Eastlake, D., "Domain Name System Security Extensions",
               RFC 2065, March 1999.

   [FTP-IPV6] Allman, M., Ostermann, S. and C. Metz, "FTP Extensions for
               IPv6 and NATs", RFC 2428, September 1998.

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

   [NAT]   Egevang, K. and P. Francis, "The IP Network Address
              Translator (NAT)", RFC 1631, May 1994.

   [NAT-TERM] Srisuresh, P. and M. Holdrege, "IP Network Address
              Translator (NAT) Terminology and Considerations", RFC
              2663, August 1999.

   [SIIT]   Nordmark, E., "Stateless IP/ICMP Translator (SIIT)", RFC
              2765, February 2000.

   [TRANS]   Gilligan, R. and E. Nordmark, "Transition Mechanisms for
              IPv6 Hosts and Routers", RFC 1933, April 1996.

   [V6ADDR]   Hinden, R. and S. Deering, "IP Version 6 Addressing
              Architecture", RFC 2373, July 1998.

Authors' Addresses

   George Tsirtsis
   Internet Futures
   B29 Room 129
   BT Adastral Park
   IPSWICH IP5 3RE
   England

   Phone: +44 181 8260073
   Fax: +44 181 8260073
   EMail: george.tsirtsis@bt.com
   EMail (alternative): gtsirt@hotmail.com

   Pyda Srisuresh
   630 Alder Drive
   Milpitas, CA 95035
   U.S.A.

   Phone: (408) 519-3849
   EMail: srisuresh@yahoo.com

Full Copyright Statement

Acknowledgement