Network Working Group   B. Fink
Request for Comments: 2921   ESnet
Category: Informational   September 2000

6BONE pTLA and pNLA Formats (pTLA)

Status of this Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

Abstract

This memo defines how the 6bone uses the 3FFE::/16 IPv6 address prefix, allocated in RFC 2471, "IPv6 Testing Address Allocation", [6BONE-TLA], to create pseudo Top-Level Aggregation Identifiers (pTLA's) and pseudo Next-Level Aggregation Identifiers (pNLA's).

Acknowledgements

The address formats here are contributions of various early participants of the 6bone testbed project, and of the IPng and NGtrans IETF working groups.

Table of Contents

1. Introduction

This memo defines how the 6bone uses the 3FFE::/16 IPv6 address prefix, allocated in RFC 2471 [6BONE-TLA], to create pseudo Top-Level Aggregation Identifiers (pTLA) and pseudo Next-Level Aggregation Identifiers (pNLA).

The guiding specifications for IPv6 addressing relating to the 6bone prefix, and the pTLA and pNLA formats, are "IP Version 6 Addressing Architecture" [ADDRARCH], and "An IPv6 Aggregatable Global Unicast Address Format" [AGGR].

The purpose of creating pseudo TLA and NLA formats for the 6bone is to provide a prototype of the actual TLA and NLA formats as they might be used in production IPv6 networks. To do this economically, using only a minimum of real production IPv6 address space, a single TLA, 3FFE::/16, was reserved by the IANA (Internet Assigned Numbers Authority) for testing on the 6bone. Thus it was necessary to define a pretend-to-be, or pseudo, TLA and NLA structure to use under the 3FFE::/16 prefix.

Given the 48-bit length of the IPv6 Aggregatable Global Unicast Address external routing prefix (that contains the TLA and NLA identifiers), there is enough room to extend the TLA ID to contain a pTLA and shorten the NLA ID to become a pNLA. This document specifies this.

In early 1999, it was decided to change the 6bone's pTLA format to allow greater expansion of the testbed network, thus accommodating more than the original 256 pTLA-s. Thus there are now two 6bone pTLA and pNLA formats. This document specifies this.

2. 6BONE pTLA and pNLA Formats

2.1 Original 8-bit pTLA and 24-bit pNLA Format

The original pTLA and pNLA format was intended to accommodate 256 pTLA-s, i.e., backbone networks carrying IPv6 transit traffic.

The original TLA and NLA ID-s as specified in [AGGR] are as follows:

```
| 3 |  13 |          32          |   16   |    64 bits      |
+---+-----+----------------------+--------+-----------------+
|001| TLA |       NLA ID         | SLA ID | Interface ID    |
+---+-----+----------------------+--------+-----------------+
```

The TLA value 1FFE was assigned to the 6bone, which when viewed with the 3-bit format prefix in prefix notation form is 3FFE::/16.

The first 8-bits of the NLA ID space are assigned as the pTLA that defines the top level of aggregation (backbone) for the 6bone. This provides for 256 6bone backbone networks, or pTLA-s, and leaves a 24-bit pNLA ID for each pTLA to assign as needed.

```
|     16    |  8  |      24      |   16   |    64 bits      |
+-+---------+-----+--------------+--------+-----------------+
|  0x3FFE   |pTLA |     pNLA     | SLA ID | Interface ID    |
+-+---------+-----+--------------+--------+-----------------+
```

In prefix notation form the pTLA is 3FFE:nn00::/24, where nn is the pTLA assignment.

The remaining NLA ID space can be used by each pTLA for their downward aggregated delegation:

```
|  n  |      24-n bits     |   16   |    64 bits      |
+-----+--------------------+--------+-----------------+
|pNLA1|       Site         | SLA ID | Interface ID    |
+-----+--------------------+--------+-----------------+

      |  m  |    24-n-m    |   16   |    64 bits      |
      +-----+--------------+--------+-----------------+
      |pNLA2|    Site      | SLA ID | Interface ID    |
      +-----+--------------+--------+-----------------+

            |  o  |24-n-m-o|   16   |    64 bits      |
            +-----+--------+--------+-----------------+
            |pNLA3|  Site  | SLA ID | Interface ID    |
            +-----+--------+--------+-----------------+
```

The pNLA delegation works in the same manner as specified in [AGGR]. pTLA's are required to assume registry duties for the pNLA's below them, pNLA1's for those below them, etc.

2.2 New 12-bit pTLA and 20-bit pNLA Format

After it became clear that the 6bone would become a useful testbed for transition, in addition to its early role as a testbed for specifications and implementations, the 6bone community decided to expand the size of the pTLA ID.

Several important decisions regarding this expansion of the pTLA field are:

1. to leave the currently allocated 8-bit pTLA-s in use until the space was needed, thus relying on a range value check to indicate the new pTLA format,

2. to use a modulo 4-bit sized pTLA ID to make reverse path entry into the DNS easier,

3. given 2. above, to keep the pTLA ID size as small as possible to not restrict pNLA ID size.

Therefore, the first 12-bits of the NLA ID space are assigned as the pTLA that defines the top level of aggegation (backbone) for the 6bone. This would eventually provide for 4096 6bone backbone networks, or pTLA-s, and leaves a 20-bit pNLA ID for each pTLA to assign as needed.

```
|     16     |   12  |    20     |   16   |     64 bits      |
+-+---------+-------+-----------+--------+-----------------+
|  0x3FFE   | pTLA  |   pNLA    | SLA ID | Interface ID    |
+-+---------+-------+-----------+--------+-----------------+
```

In prefix notation form the pTLA is 3FFE:nnn0::/28, where nnn is the pTLA assignment. However, as the existing 8-bit pTLA's are being left in use for the present, the nnn value starts at 0x800 for now, thus yielding only 2048 pTLA's in this new format.

The remaining NLA ID space can be used by each pTLA for their downward aggregated delegation:

```
|  n  |      20-n bits     |   16   |     64 bits      |
+-----+--------------------+--------+-----------------+
|pNLA1|       Site         | SLA ID | Interface ID    |
+-----+--------------------+--------+-----------------+

      |  m  |    20-n-m    |   16   |     64 bits      |
      +-----+--------------+--------+-----------------+
      |pNLA2|    Site      | SLA ID | Interface ID    |
      +-----+--------------+--------+-----------------+

            |  o  |20-n-m-o|   16   |     64 bits      |
            +-----+--------+--------+-----------------+
            |pNLA3|  Site  | SLA ID | Interface ID    |
            +-----+--------+--------+-----------------+
```

As with the original pTLA format, the pNLA delegation works in the same manner as specified in [AGGR]. pTLA's are required to assume registry duties for the pNLA's below them, pNLA1's for those below them, etc.

2.3 Example Format For pNLA's

An example usage of the pNLA space is given to demonstrate what is reasonable and possible. It should not be assumed that this implies the pNLA space must be used this way. As the new pTLA and pNLA format is now the default, the example here assumes the 20-bit pNLA format.

The following example provides for up to 255 intermediate transit ISP's (called pNLA1 below). The pNLA1 value of zero is meant to indicate that there is no intermediate transit ISP between the backbone pTLA network and the end user site.

```
|<-----20-bit pNLA ID----->|
|                          |
|  8  |       12 bits      |   16   |    64 bits      |
+-----+--------------------+--------+-----------------+
|pNLA1|      Site  ID      | SLA ID | Interface ID    |
+-----+--------------------+--------+-----------------+
```

Intermediate transit networks (pNLA1's) would assign uniques Site ID's for eachend user site served.

As an example of this, assuming a backbone pTLA of 0x800, no intermediate transit ISP (thus a pNLA1 of 0x00) and a sequential site ID (with start at the right edge numbering) of 0x0001, the routing prefix for the first site would look like:

```
         3FFE:8000:0001/48
  6bone _|||| |||| ||||___site
              |||| |
  b/b site____|||| |
                 | |
  transit________|_|
```

Another example of this usage, assuming the same backbone pTLA1 of 0x800 and an intermediate transit ISP under it (numbering from the left edge) with an NLA1 of 0x80, and a sequential site ID of 0x0001, the routing prefix for the first site connected would look like:

```
         3FFE:0180:0001/48
  6bone _|||| |||| ||||___site
              ||||
  b/b site____||||
                ||
  transit_______||
```

Note 1: the two sites numbered 0x001 in the above examples are really two different sites as their pNLA1 authority above them is different (i.e., in the first case no transit exists thus the site is directly connected to the pTLA backbone ISP, and in the second case the site is directly connected to intermediate transit ISP 0x80).

Note 2: there would be nothing to prevent an pNLA1 transit site from further allocating pNLA's below, but that becomes the policy of the pTLA and pNLA's above them to work out.

Note 3: The 6bone registry, which is a RIPE-style database for documenting IPv6 sites connected to the 6bone, has an "inet6num" object to allow documentation of all IPv6 addresses allocated.

3. Security Considerations

IPv6 addressing documents do not have any direct impact on Internet infrastructure security.

References

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

[AGGR] Hinden, R., O'Dell, M. and S. Deering, "An IPv6 Aggregatable Global Unicast Address Format", RFC 2374, July 1998.

[HARDEN] Rockell, R. and R. Fink, "6Bone Backbone Routing Guidelines", RFC 2772, February 2000.

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

[6BONE-TLA] Hinden, R., Fink, R. and J. Postel, "IPv6 Testing Address Allocation", RFC 2471, December 1998.

Author's Address

Bob Fink, ESnet
Lawrence Berkeley National Lab
MS 50A-3111
1 Cyclotron Road
Berkeley, CA 94720
USA

Phone: +1 510 486 5692
Fax: +1 510 486 4790
EMail: fink@es.net

Full Copyright Statement

Acknowledgement

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