Network Working Group
Internet Draft
Expiration Date: September 1999

R. Elz
University of Melbourne

March 1999

The Process of Renumbering

draft-ietf-ipngwg-renum-process-00.txt

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

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.

To view the list Internet-Draft Shadow Directories, see http://www.ietf.org/shadow.html.

Note that the first paragraph of this section is a meaningless bureaucratic requirement of the IESG. It is provided so as to satisfy those bureaucratic requirements, and serves no other purpose whatever. Information as to any intellectual property rights, beyond the right to redistribute this document and make use of it for the purposes of an internet draft, should be sought in other parts of this document.

This entire section has been prepended to this document automatically during formatting without any direct involvement by the author(s) of this draft. No assumption should be made that the authors have assented to any of it.

Abstract

This document discusses the process of renumbering an Internet Node. While it is not protocol specific, as such, it is expected that some of the mechanisms proposed will never be defined, or deployed, for IPv4. Hence this may serve mostly as a template mechanism for the process of renumbering an IPv6 site.

1. Introduction

Much has been written [...] on the subject of renumbering a site, and its difficulties. This document does not consider that issue, and assumes that the results of the lessons learned from those experiences, and the benefits of IPv6 including stateless autoconfiguration via neighbour discovery [] and router renumbering [] will make the task of renumbering a site, internally, a tractable problem.

However, of itself, this is insufficient to fully complete a renumbering task. Aside from the site itself, there is the problem of others, typically unknown to the site itself, who have knowledge of its old address, and need to be informed of any address changes and then make the necessary updates.

This document describes the a series of steps which, if followed, will permit all parties on the internet with a need to know of address changes to detect the change, and update their knowledge, in such a way that the address change can be made gracefully.

2. The Steps

The overall process of renumbering can be described by the following sequence of events:

[1] Someone/something determines that a new set of addresses will be needed for some set of hosts/nets. The details of how, and by whom, this decision is made are beyond the scope of this document.

[2] That info is communicated to the owner of (the net admin, human or otherwise, of) the set of hosts involved. Whether this is automated via some protocol or other, or is done via more mundane method, is not important here.

[3] Some kind of DNS (or some kind of database) entry is updated showing the new set of addresses as additional addresses for the entity involved. This dadatase does not yet exists. Its creation seems to be necesary for orderly

renumbering.

[4] Routers and other filter lists (etc) are updated to make the new addresses equivalent to the old ones (all around the internet, not just at the entity being renumbered.) This presumes that the routers, or the manager or configuration system that configures the routers, use keys into the database postulated in the previous step when configuring references to address spaces, and translate those to the corresponding numeric values as needed, along with "time to live" values to cause regular retranslation of the keys to address values.

[5] The new addresses are distributed to the hosts/routers that are being renumbered. The precise mechanism by which this is accomplished is unimportant here, however it seems likely that [Router-renum] might be used to distribute the new addresses to routers, followed by appropriate Router Advertisments [RFCmmm] to distribute the new addresses to the hosts. The old addresses should be marked as deprecated at the same time. This will cause those hosts to start originating connections from the new addresses one assumes.

[6] The DNS is updated with the new addresses for the things that have been renumbered. This will gradually cause incoming connections to start using the new addresses as new translations of names to addresses are done, and the old cached address values gradually disappear.

[7] The old addresses are removed from the DNS so no new connections will be initiated to the old addresses.

[8] The old addresses are withdrawn from RA adverts, so the renumbered hosts stop accepting and using them.

[9] The database mentioned in 3 is updated to remove the old addresses from the list of addresses belonging to the entity that has been renumbered.

[10] Step 4 naturally repeats as time to lives expire, with the translation of keys to addresses now using only the new addreses.

[11] The address block that is no longer in use is returned to the free pool and is thus now available for assignment elsewhere.

3. Time Lines

The steps of the previous section must be completed in order. However, some require delays to allow proper propogation of changed information around the internet.

The database mentioned in step 3 needs to associate a time to live with each value. Then steps 4 and 10 need to be given at least that time to complete. Note that if the DNS implements the database, the relevant time for steps 4 and 10 is the sum of the TTL field of the DNS for the resource records used, and the zone propogation time from primary server to secondary servers, as set in the Start of Authority resource record. Typically an implementation of this scheme would allow twice the expected necessary delay.

The delay required at step 5 will depend upon the size of the part of the network being renumbered. As this is entirely confined to the site being renumbered, it can determine when it has finished.

Step 6 can proceed concurrently with step 5, to the extent that as each node gains its new addreses it can immediately enter them in the DNS. This permits the use of techniques like [DynDNS] to allow DNS updates to be performed by the renumbered hosts.

Similarly, step 7 can proceed concurrently with step 6, in that old addresses can be deleted from the DNS as the new ones are added, if desired. After the last DNS update of step 7 is made, that is, the last of the old addresses is deleted from the DNS, the minimum delay before step 8 is the DNS zone propogation and cache TTL delay. Typically a much longer delay will occur at this point to allow connections already established to remain operational. Until the connection identifiers are separated frin the addresses, or a mechanism is created to allow the connection identifiers to be altered during the life of a connection, a considerable delay is likely to be required here. In gthe worst cases this delay could amount to months, or even years. Of course, it will be bounded by the period during which the old addresses will be permitted to remain. When that time expires, and assuming the additional mechanism does not exist, old connections will simply have to be broken.

Step 8 is essentially a repeat of step 4, and should take an equivalent period. Similarly, step 9 uses the same procedures as step 5.

As soon as step 11 is completed, the whole procedure has finished.

4. Requirements

5. Security Considerations

6. Example

7. References

Authors' Addresses