Network Working Group
Internet-Draft
Expires: March 22, 2007

T. Chown
M. Thompson
A. Ford
S. Venaas
University of Southampton, UK

September 18, 2006

Things to think about when Renumbering an IPv6 network draft-chown-v6ops-renumber-thinkabout-05

Status of this Memo

Copyright Notice

Abstract

This memo presents a summary of scenarios, issues for consideration and protocol features for IPv6 network renumbering, i.e. achieving the transition from the use of an existing network prefix to a new prefix in an IPv6 network. Its focus lies not in the procedure for renumbering, but as a set of "things to think about" when undertaking such a renumbering exercise.

Table of Contents

1. Introduction

This memo presents a summary of scenarios, issues for consideration and protocol features for IPv6 network renumbering, i.e. achieving the transition from the use of an existing network prefix to a new prefix in an IPv6 network. This document does not relate the procedures for IPv6 renumbering; for such a procedure the reader is referred to [1]. The authors plan to use this document, together with ongoing operational experience, to refine [1] where necessary, to promote that guide from Informational to BCP. The focus is on renumbering site networks, though many of the principles apply to renumbering other (ISP) networks.

1.1. Structure of Document

This document is split into a number of sections that discuss various aspects of network renumbering that should be considered when undertaking such an event. This document begins with a discussion of the various reasons behind renumbering events, and the requirements to ensure the event goes smoothly. The following sections then discuss a selection of factors that can both help and hinder the renumbering procedure, and as such should be taken into account when planning the event. Finally, this document summarises issues with applications and services, and attempts to identify places where IP addresses may be hard-coded and thus require reconfiguration during a renumbering event.

1.2. Past IPv4 Renumbering studies in the PIER WG

A number of years ago (1996-1997), the Procedures for Internet/ Enterprise Renumbering (PIER) WG spent time considering the issues for IPv4 renumbering. The WG produced three RFC documents. In RFC1916 [2], a "call to arms" for input on renumbering techniques was made. RFC2071 [3] documents why IPv4 renumbering is required. Interestingly, many, but not all, of the drivers have changed with respect to IPv6. In RFC2072 [4], a Router Renumbering Guide, some operational procedures are given, much as they are in Baker [1] for IPv6.

Reflection on RFC2071 is interesting, witness the quote: "It is also envisioned that Network Address Translation (NAT) devices will be developed to assist in the IPv4 to IPv6 transition, or perhaps supplant the need to renumber the majority of interior networks altogether, but that is beyond the scope of this document." That need however is still very strong, particularly given the lack of Provider Independent (PI) address space in IPv6 (in IPv4, PI address space exists mainly for historical, pre-CIDR reasons).

RFC2072 is more interesting in the context of this document. Some is certainly relevant, though much is not, due to the inherent changes in IPv6. For example, there is no CIDR and address aggregation is given as mandate. Also, IPv6 subnets are in effect fixed length (/64), so local administrators do not need to resize subnets to maximise efficient use of address space as they do in IPv4.

One core message from RFC2072 that holds true today is that of section 4 where the observation is made that renumbering networks whilst remaining the same hierarchy of subnets (i.e. the cardinality of the set of prefixes to renumber remains constant) is the 'easiest' scenario to renumber; when each "old" prefix can be mapped to a single "new" prefix.

A distinction of this work is that, where the PIER working group consider the transition from IPv4 to IPv6 addressing as a renumbering scenario, we strictly consider only the renumbering from IPv6 prefixes to other IPv6 prefixes and leave transition to well documented techniques such as those from the PIER working group.

2. Terminology

The following terminology is used in this document (to be expanded in future revisions):

- Site: An organisationally distinct network, ranging from SOHO through to enterprise.
- Flag day: A planned service outage.
- Node: A device on the network that is being renumbered, or that is involved in communication with the network being renumbered.

3. Renumbering Event Triggers

This section details typical actions that result in the need for a renumbering event, and thus define the scenarios for renumbering.

In many instances, in particular those where no "flag day" is involved, the process of renumbering will inevitably lead to a scenario where hosts are multi-addressed or multi-homed as one phase of the renumbering procedure. The relationship between renumbering and multi-homing is discussed later in this document.

In other instances, e.g. a change in the IPv4 address offered by a provider to a site using 6to4 [9], the change offers no overlap in

external connectivity or addressing, and thus there is no multi-homing overlap.

Triggers may be provider-initiated or customer-initiated.

Triggers and scenarios for IPv4 renumbering are discussed in RFC2071, but many of these are no longer relevant, and in IPv6 some new triggers exist, e.g. those related to network mobility or IPv6 transition tools.

3.1. Change of uplink prefix

One of the most common causes for renumbering will be a change in the site's upstream provider. As per RFC3177 [10], the typical allocation for an IPv6 site is a /48 size prefix taken from the globally aggregated address space of the site's provider. With IPv6, sites are highly unlikely to be able to obtain provider independent (PI) address space, as have in some cases been obtained in the past with IPv4. Rather, sites use provider assigned (PA) addressing. As a result, if a site changes provider, it must also change its IPv6 PA prefix.

3.1.1. Migration to new provider

In the simplest case, the customer is triggering the renumbering by choosing to change the site's upstream provider to a new ISP and thus a new PA IPv6 prefix range. This may simply be in the form of selecting a new commercial provider, although there are several other possible scenarios, such as changing from a dial-up to a broadband connection, or moving from a community wireless connection to a fixed broadband connection.

A similar scenario exists when a customer migrates to a different service from the same provider. For example, if a customer changes from a dialup to a broadband connection, they will likely be connecting to a different part of the provider's topology, and therefore receive a different address allocation.

3.1.2. Dial on Demand

A site may connect intermittently to its upstream provider. In such cases the prefix allocated by the provider may change with each connection, as it often does in the case of single IPv4 address allocations to SOHO customers today. Thus the site may receive a prefix still in its provider PA range, but the prefix may vary with each connection, causing a renumbering event.

Dynamically assigned IP addresses are common today with dial-up and

ISDN Internet connections, and to a lesser extent some broadband products, particularly cable modems. Usually with dynamically assigned IP addresses on broadband products, the address is only likely to change when the customer reconnects, which could be very infrequently.

This case can be mitigated by encouraging ISPs to offer static IPv6 prefixes to customers. Where /48 prefixes are provided, a large ISP may be forced to require significantly more than the "default" /32 allocation from an RIR to an ISP to be able to service its present and future customer base. With always-on more common in new deployments, provider re-allocation should be less common; however the practice of reallocating IPv4 addresses in SOHO broadband networks is not uncommon in current broadband ISPs.

3.1.3. Provider migration and upstream renumbering

A site's upstream provider may need to renumber, due for example to a change in its network topology or the need to migrate to a different or additional prefix from its Regional Internet Registry (RIR). This will in turn trigger the renumbering of the end site.

Such renumbering events would be expected to be rare, but it should be noted that RIR-assigned IPv6 address space is not owned by an ISP.

3.1.4. IPv6 transition

During transition to IPv6, there are several scenarios where a site may have to renumber. For example, if the site uses 6to4 for access and its IPv4 address is dynamically assigned, an IPv6 renumbering event will be triggered when the site's IPv4 address changes.

Another likely renumbering event would be the change of transition mechanism, such as from 6to4 to a static IPv6-in-IPv4 tunnel, or from any one of those mechanisms to a native IPv6 link. When changing from 6to4 (2002::/16) addresses to native global aggregatable unicast addresses, renumbering would be unavoidable. When migrating from a tunnelled to a native connection, renumbering may not be necessary if the same prefix can be routed natively, however this would be provider-dependent.

In addition, there are likely to be many cases of network renumbering occurring when the old 6bone prefix (3FFE::/16) is phased out as per RFC3701 [11], and networks still using it will have to renumber.

Finally, there is at least one transition mechanism, ISATAP [12], that uses specially crafted host EUI-64 format addresses. Should a site migrate from ISATAP to use either conventional EUI-64 addressing

(via stateless address autoconfiguration or perhaps DHCPv6), then renumbering would be required at least in the host part of addresses.

It is also worth noting that nodes that use IPv6 Privacy Extensions [13] will in effect renumber the host part of their address on a frequent basis, in the case of one popular implementation on a daily basis if the node remains on-link on the same network.

3.2. Change of internal topology

A site may need to renumber all or part of its internal network due to a change of topology, such as creating more or less specific subnets, or acquiring a larger IPv6 address allocation. Motivations for splitting a link into separate subnets may be to meet security demands on a particular link (policy for link-based access control rules), or for link load management by shuffling popular services to more appropriate locations in the local topology. Link-merging may be due to department restructuring within the hosting organisation, for example.

3.3. Acquisition or merger

Two networks may need to merge to one due to the acquisition or merger of two organisations or companies. Such a reorganisation may require one or more parts of the new network to renumber to the primary PA IPv6 prefix.

3.4. Network growth

A site that is allocated a /48 prefix may grow to a size where it needs to use a larger prefix for internal networking. Sites in the early stages of IPv6 deployment may only request a /48, even if they are likely to outgrow such a prefix in time. In such a case site-wide renumbering may be required to utilise the new prefix if organisational restructuring also happens due to the growth.

3.5. Network mobility

This covers various cases of network mobility, where a static or nomadic network may obtain different uplink connectivity over time, and thus be assigned different IPv6 PA prefixes as the topology changes. One example is the "traditional" NEMO network [14], another may be a community wireless network where different sets of nodes gain uplink connectivity - typically to the same provider - at different times.

4. Renumbering Requirements

In this section we enumerate potential specific goals or requirements for sites or users undergoing an IPv6 renumbering event.

4.1. Minimal disruption

The renumbering event should cause minimal disruption to the routine operation of the network being renumbered, and the users of that network.

Disruption is a difficult term to quantify in a generic way, but it can be expressed by factors such as:

- Application sessions being terminated
- Security controls (e.g. ACLs) blocking access to legitimate resources
- Unreachability of nodes or networks
- Name resolution, directory and configuration services providing invalid (out-of-date) address data
- Limitation of network management visibility

These disruptive elements will be covered in situ as we discuss protocol features and other renumbering considerations later in this memo.

4.2. Session survivability

The concept of session survivability is catered for by [1] in that new sessions adopt either old or new prefix based on the state of the renumbering process, as discussed in Section 5.1. However, other approaches to renumbering networks may be appropriate in certain deployments, such as where "flag days" are unavoidable, such as where two live prefixes are being "swapped". In these cases, further consideration for existing sessions (their longevity, frequency, independence across interactions, etc.) is required.

Some protocols are specifically geared to aid session survivability, e.g. the Stream Control Transmission Protocol (SCTP) [15], and may prove valuable in mission-critical renumbering scenarios, in particular the extension that enables the dynamic addition and removal of IP addresses from an SCTP endpoint association [16].

Sessions may be administratively maintained, such as NFS mounts for

user filestore, or they may be user-driven, e.g. long-running ssh sessions.

In general, it is important to consider how TCP and the applications above it handle the connection failures that may result from a change in address.

There are different classes of session duration, as described in the following sections.

4.2.1. Short-term session survivability

A typical short-term session would involve a request-response protocol, such as HTTP, where a new network connection is initiated per transaction, or at worst for a small transaction set. In such cases the migration to a new network prefix is transparent: the client can use the new prefix in new transactions without consequence. Some applications, however, may be skewed by such a shift in connection source for the same entity 'user', for example applications that use recent connection history as a cue to identity (e.g. POP-before-SMTP as used by many dial-on-demand ISP customers <http://popbsmtp.sourceforge.net/>), or for applications that care about connection statistics (the same user web-browsing "session" may be split into two where a renumbering event occurs in-between client transactions).

4.2.2. Medium-term session survivability

A medium-term session is typified by an application or service that may persist for perhaps a period of a few minutes up to a period of a day or so. This might involve a TCP-based application that is left running during a working day, such as an interactive shell (SSH) or a large file download.

4.2.3. Long-term session survivability

Long term sessions may typically run for several days, if not weeks or months. These might typically include TCP-based NFS mounts, or long-running TCP applications. Sessions in this context may also include those applications that, once started, do not re-resolve names and so repeatedly open new connections or send new datagrams to the same (as bound at time of initialisation) address throughout their execution lifetime. Even if at API-level applications do attempt to re-resolve the symbol to which they desire to connect, the behaviour of the resolvers is unclear as to whether mappings are refreshed from the naming service, and as such even if the renumbering site does update its DNS (or NIS, LDAP database etc.), the local result may indeed be cached without any indication passed

back up to the application as to how 'old' said binding information is.

4.2.4. "Sessions" in non-session based transports

UDP transport protocols, such as UDP-based NFS mounts, maintain the status of a 'session' by keeping state at one or both ends of the communication, but without a persistent open socket connection at the network layer. If, due to node renumbering, one endpoint changes address then that state becomes invalid and the 'session' interrupted.

Note that some stack implementations do not correctly flag an error to applications that attempt to send packets with an invalidated source address, see section Section 9.5

IP addresses are also seen carried in higher-layer protocols, e.g. application sessions, such as with FTP. Any application that makes use of layer-3 address data as a unique end-point identifying token may be disrupted by the address of the node changing to which that token relates. This may not be an issue in cases where the token is treated as abstract (i.e. literally just a token), however where locator semantics are inferred, subsequent attempts to 'resolve' the token to an address endpoint for communication, for example, will fail.

5. IPv6 Protocol Features and their Effects on Renumbering

IPv6 includes a number of notable features that can help or hinder - and sometimes both - renumbering episodes. This section discusses these features and their associated effects for consideration when undertaking network renumbering, both in terms of how they can be used to ease the process, as well as potential pitfalls that should be considered.

5.1. Multi-addressing

As per RFC3513 [17], IPv6 hosts may be multi-addressed. This means that multiple unicast addresses can be assigned and active on the same interface. These addresses can have different reachabilities ('scopes' such as link-local or global), different statuses including 'preferred' and 'deprecated', and may be ephemeral in nature (such as care-of addresses when attached to a foreign network [18] or IPv6 Privacy addresses [13]). RFC3484 address selection semantics [5] determine which of the "MxN" address pairs to use for communication in the general case.

During a renumbering episode, the addition of an extra address for an endpoint increases the number of possible source-destination address pairs for communications between nodes to use. The address selection mechanisms specified by RFC3484 are currently at varying stages of implementation in operating systems.

RFC3484 also specifies policy hooks to allow administrative override of the default address selection behaviour, for example to specifically prefer a source prefix for use with a set of particular destinations. It is thought that this policy-based address selection may be of benefit in renumbering events, or used in the development of bespoke renumbering tools.

Multi-addressing also creates various issues with border filtering, discussed in detail in Section 7.2.

5.2. Multi-homing techniques

A multi-homed site is a site which has multiple upstream providers. A site may be multi-homed for various reasons, however the most common are to provide redundancy in case of failure, to increase bandwidth, and to provide more varied, optimal routes for certain destinations.

In renumbering, multi-homing will either be a temporary state, during the transition, or be a permanent feature of the network configuration, which may be being altered during the renumbering.

5.2.1. Relevance of multi-homing to renumbering

As discussed in section 2, and in particular section 2.5, of [1], during the 'without a flag day' renumbering procedure there will be a period where both the old and the new prefixes are stable and valid for the network. During such a period, the network may be multi-homed, and as such many of the issues relating to multi-homing in IPv6 are also relevant, albeit in a small capacity, to the renumbering procedure. A stable multi-homed situation must therefore be a requirement for renumbering without a 'flag day'.

In such a situation, however, the multi-homed state will not be permanent, and will only exist for the duration for which it is required, i.e. for the period during the renumbering procedure when both prefixes should be valid.

Renumbering can also occur, however, in a network that is already multi-homed, for example with redundant links to multiple providers. Such a site may wish to renumber for any of the situations given in the earlier section, as well as renumbering because of changes in the

number of upstream providers. If at least one of the upstream links remains unchanged during the renumbering, however, then these links could be used exclusively for that period, alleviating some of the issues with prefix changes. The stable link(s) could therefore be the only prefixes advertised as valid for the 'stable state', with the removal of the old prefix and introduction of the new prefix being separate events.

Until the best practice for the multi-homing situation is defined, however, its effect on renumbering is not a focus of this document.

5.2.2. Current situation with IPv6 multi-homing

Unlike IPv4 multi-homing, where PI address space is relatively easy to obtain and thus a site can broadcast its own routing information, most IPv6 addresses will be PA addresses and thus the site will have no control over routing information. Multi-homing in IPv6 therefore does not necessarily exist in the same way as in IPv4 and the multi6 [38] working group was chartered to try to find a solution.

Most IPv6 multi-homing solutions fall into the categories of either being host-centric, where it is the hosts that are multi-addressed, and choose which addresses to use, or site-based, where it is the site exit routers that decide which connections to use. The simplest solutions are extensions of the current multi-addressing techniques, but these suffer from the problem that, at some point, connections using the old addresses will be broken.

The more advanced solutions [19], and in particular the solution taken forward into the shim6 [39] working group, examine the potential for splitting the 'identity' and 'location' features of IP, currently both represented by the IP address, and connecting to a host's identity, rather than its address, so that connections can continue unhindered across renumbering events. Such solutions are, however, very much in their infancy and as yet do not provide a stable solution to this problem.

Support for the level of multi-homing required during a renumbering exercise is, however, mostly provided by multi-addressing (Section 5.1), since all that is primarily required is stable use of either prefix for a given period. The core issue remains, however, that at some point the connections using the old address will be broken when the addresses are removed. The impact of this can be limited as best as possible during the renumbering procedure.

5.3. Mobile IPv6

Mobile IPv6 (MIPv6) [18] specifies routing support to permit an IPv6

host to continue using its "permanent" home address as it moves around the Internet. Mobile IPv6 supports transparency above the IP layer, including maintenance of active TCP connections and UDP port bindings. There are a number of issues to take into account when renumbering episodes occur where Mobile IPv6 is deployed:

Renumbering a network which has mobile IPv6 active is a potentially complex issue to think about. In particular, can changed router advertisements correctly reach the mobile nodes, and can they be correctly renumbered, like a node on the local network? In addition, an even more complex issue is what happens when the home agent renumbers? Is it possible for the mobile nodes to be informed and correctly renumber and continue, or will the link be irretrievably broken?

5.3.1. Visited site renumbers when mobile

When a node is mobile and attached to a foreign network it, like any other node on the link, is subject to prefix renumbering at that site. Detecting a new prefix through the receipt of router advertisements, the mobile node can then re-bind with its home agent informing it of its care-of address - just as if it had detached from the foreign network and migrated elsewhere. Where the node receives forewarning of the renumbering episode, the Mobility specification suggests that the node explicitly solicits an update of the prefix information on its home network

5.3.2. Home site renumbers when mobile

When mobile, a host can still be contacted at its original (home) address. Should the home network renumber whilst the node is away but active (i.e. having bound to the home agent and registered a live care-of address), then it can be informed of the new global routing prefix used at the home site through the Mobile Prefix Solicitation and Mobile Prefix Advertisement ICMPv6 messages (sections 6.7 and 6.8 of RFC3775 [18] respectively).

5.3.3. Home site renumbers when disconnected

Finally, if a mobile node is detached (i.e. no binding with the home agent exists with the node present on a foreign network) and the home network renumbers, the recommended procedure - documented as an appendix to the mobility specification and therefore not necessarily proven - is to fall back to alternative methods of 'rediscovering' its home network, using the DNS to find the new global routing prefix for the home network and therefore the Home Agent's subnet anycast address, 'guessing' at what the node's new home address would be on the basis of a 64 bit prefix and 64 bit interface identifier, and

then attempting to perform registration to bind its new location.

5.4. Multicast

IPv6 supports an enriched model of multicast compared to IPv4 in that there are well-defined scopes for multicast communication that are readily expressed in the protocol's addressing architecture. Multicast features much more prominently in the core specification, for example it is the enabling technology for the Neighbour Discovery protocol (a much more efficient approach to layer 2 address discovery than compared to ARP with IPv4).

Where multicast is used to discover the availability of core services (e.g. all DHCPv6 servers in a site will join FF05::1:3), the effect of renumbering the unicast address of those services will mean that the services are still readily discoverable without resorting to a (bespoke or otherwise) service location protocol to continue to function - particularly if (unicast) ULAs are not deployed locally as per Section 5.5.

One issue related to IPv6 multicast and renumbering is the embedding of unicast addresses into multicast addresses specified in RFC3306 [20] and the embedded-RP (Rendezvous Point) in RFC3956 [21].

The former is purely a way of assigning addresses that helps with multicast address assignment, avoiding different sites from using the same multicast addresses. If a site's unicast prefix changes, then one will also need to change the multicast addresses. By way of example, a site renumbering away from prefix 2001:DB8:BEEF::/48" might have globally-scoped multicast addresses in use under the prefix "FF3E:30:2001:DB8:BEEF::/96". One may continue using the old addresses for a while, but this should be avoided since another site may inherit the prefix and they may end up using the same multicast addresses.

The issue with embedded-RP is that, by definition, the RP address is embedded. So if the RP address changes, then the group addresses must also be changed. This may happen not only when a site is renumbered, but also if a site is restructured or the RP is moved within the site. The embedded address is used by routers to determine the RP address. Applications must use new group addresses once the RP is not available on the old address.

Another interesting topic is multicast source renumbering. With traditional multicast a source should be able to start streaming from a new address, and nodes belonging to the multicast group will immediately start receiving. There might be some application issues though. If sources are identified by the source address only, then

this might appear as a new source to the receivers (as they would where IPv6 Privacy addresses are used). Using RTP a receiver may determine it's the same source.

With Source Specific Multicast (SSM), source renumbering is more complicated since receivers must specify exactly which sources they want to to receive from. This means that receivers must somehow be told to join the new source addresses, and must be able to discover those addresses.

5.5. Unique Local Addressing

Section 5 of [22] suggests that the use of Local IPv6 addresses in a site results in making communication using these addresses independent of renumbering a site's provider based global addresses. It also points out that a renumbering episode is not triggered when merging multiple sites that have deployed centrally assigned unique local addresses[23] because the FC00::/7 ULA prefix assures global uniqueness. The use of ULAs internally should ideally mitigate against global address renumbering of nodes, particularly as intra-site communication can continue unhindered by the change in global address prefixes due to provider migration or re-assignment of prefix from an upstream.

ULAs appear to lend themselves particularly well for long-lived sessions (from the categorisation Section 4.2.3) whose nature is intra-site, for example local filestore mounts over TCP-mounted NFS: With clients using ULA source addresses to mount filestore using the ULA of an NFS server, both client and server can have their global routing prefix renumbered without consequence to ongoing local connections.

When merging two sites that have both deployed FC00::/7 locally-assigned ULA prefixes, the chance of collision is inherently small given the pseudo-random global-ID determination algorithm of [22]. Consideration of possible collisions may be prudent however unlikely the occurrence may be.

With reference to section 2 of [1], the adoption of ULA to assist in network renumbering can be considered a 'seasoning' of Baker's renumbering procedure: where interaction between local nodes and their services cannot suffer the inherent issues observed when migrating to a new aggregatable global unicast prefix, the use of FC00::/7 unique local addresses may offer an appropriately stable and reliable solution. Whilst on the surface, the use of ULAs in networks that also have global connectivity appears straightforward and of immediate benefit as regards provider migration, they currently suffer significant operational issues including address

selection, border filtering, name service provision and routing.

If addresses under a global routing prefix are deployed alongside ULAs, then nodes will need to cater for being multi-addressed with multiple addresses of the same (global) syntactic scope, e.g. follow the principles laid out in RFC3484 [5]. The administrator should ideally be able to set local policy such that nodes use ULAs for intranet communications and global addresses for global Internet communications. Note in particular that address selection policy different from the defaults of RFC3484 are required for sites that have deployed ULAs whilst making use of multicast in scopes greater than link-scope (i.e. FFx3 and higher).

5.5.1. ULAs, Multicast and Address Selection

For ordinary unicast traffic, the address selection rules of RFC3484 will function correctly. Assuming no higher-precedence rules are matched, a multi-addressed host will choose its source address through finding the address with the longest matching prefix compared with the destination address. This will pick global unicast addresses (i.e. within 2000::/3) for communication with other such addresses, and pick ULAs for other ULAs. This correct behaviour is dependent on sites running two-face DNS, however, and therefore ensuring remote sites do not know of non-routable ULAs.

The key problem with ULAs and source address selection occurs, however, when sending to multicast addresses. When it falls to the longest matching prefix tests, a ULA will always come out as preferable to a global unicast address for matching a multicast (FF00::/8) address.

This does not affect link-local multicast, however, as the preference for the appropriate scope will choose the unicast link-local address before looking at the longest prefix match (see Section 3.1 of RFC3484). For scopes wider than link-local, however, the ULA will by default always be chosen.

Local policy needs to be implemented such that, e.g., global-scope multicast addresses have the same `label' as global aggregatable unicast addresses in RFC3484 parlance. Additional rules could also be added such that site- and organisational-scope multicast addresses prefer ULAs as source addresses, again by defining an appropriate label.

Whilst no standard policy distribution mechanism exists for overriding default RFC3484 preference rules, [24] proposes the use of a DHCPv6 option in sites where stateful configuration is available.

5.5.2. ULAs with application-layer gateways

The use of ULAs may not necessarily be accompanied by provider-assigned (PA) addresses in connected networks. If addresses under a PA global routing prefix are not used, application layer gateway deployment will be required for ULA-only nodes internal to the network to communicate with external nodes that are not part of the same ULA topology.

Destination nodes that are addressed under FC00::/7 which are not part of the same administrative domain from which the ULA allocation of the local node is made, nor part of a predetermined routing agreement between two organisations utilising different ULAs for nodes within their own sites, would be filtered at the site border as usual.

Typical deployments utilising this technique would include those networks where an administrative policy decision has been made to restrict those services available to the users, or where connectivity is sufficeintly intermittent that as few nodes as possible are exposed to the issues of ephemeral connectivity.

5.6. Anycast addressing

Syntactically indistinguishable from unicast addresses, 'anycast' offers nodes a mean to route traffic toward the topologically nearest instance of a service (as represented by an IP address), relying on the routing infrastructure to deliver appropriately. RFC2526 [25] defines a set of reserved subnet anycast addresses within the highest 128 values of the 64 bit IID space. Of that space, currently only three are used, of which one is actively used and is for discovery of Mobile IPv6 Home-Agents. At the current time there are no 'global' well-known anycast addresses assigned by IANA.

In order to participate using anycast, nodes need to be configured as routers (to comply with RFC3513 [17]) and exchange routing information about the reachability of the specific anycast address. This extra level of administration requirement is negligible in the context of services as the services themselves would need configuration anyway.

There have been proposals to define globally well-known anycast addresses for core services, such as the DNS [26]. Anycast scales with regard subnet-anycast in the sense that the global routing prefix used to direct packets to an anycast node within a site is no different from any other host, and therefore nothing 'special' in the global routing architecture is required - only locally within the site does the multi-node nature of anycast need to be considered.

However, for global well-known anycast addresses to be defined, host-specific routes will need to be advertised and distributed throughout the entire Internet. As acknowledged by section 2.6 of [17], this presents a severe scaling limit and it is expected that support for global anycast sets may be unavailable or very restricted. A good discussion of best current practice for service provision using anycast addressing can be found in [27].

The use of well-known anycast addresses would assist the renumbering exercise by removing the requirement to change the addresses in the configuration of such services. The use of anycast DNS would alleviate concerns with ensuring node reconfiguration, for example when using Stateless DHCPv6 (Section 6.1.2). While anycasting datagram-based services such as DNS pose little problems, anycast does not maintain state, and so it would not be guaranteed that sequential TCP packets were to go to the same host. As discussed in [28], responses from TCP sessions begun to an anycast address should be sent from the unicast address, and future communication should continue with this address. While this means that communication will continue with the same unicast address, that address is subject to the standard address deprecation and validity. Note that anycasting of this form can be an alternative to site or organisational scope multicast service discovery as described in Section 5.4.

6. Node Configuration Issues

This section discusses how IPv6 node configuration protocols (both stateless and stateful, including DHCPv6, as well as ICMP router renumbering messages) can be used to facilitate a renumbering event, plus any complications caused by these processes, to which consideration should be given.

6.1. Stateless Address Autoconfiguration

Many IPv6 networks are likely to be configured using Stateless Address AutoConfiguration [6] (SLAAC), and in order to work through the multi-staged process as documented by Baker [1], the new prefix is introduced via router advertisements, and then the old prefix is deprecated, and finally removed.

Initially the router advertisements will contain only the prefix of the old network, then for a time they will contain both the old and the new, but with a shorter (zero) lifetime on the old prefix to indicate that it is deprecated. Finally the router advertisements will contain only the new prefix.

6.1.1. Router Advertisement Lifetimes

RFC2462 (IPv6 Stateless Autoconfiguration) [6] specifies the technique for expiring assigned prefixes and then invalidating them, such that a network has opportunity to gracefully withdraw a prefix from service whilst not terminally disrupting on-going applications that use addresses under it. Section 5.5.4 of RFC2462 in particular details the procedure for deprecation and subsequent invalidation.

By mandating as a node requirement the ability to phase out addresses assigned to an interface, network renumbering is readily facilitated: subnet routers update the pre-existing prefix and mark them as 'deprecated' with a scheduled time for expiration and then advertise (when appropriate) the new prefix that should be chosen for all outgoing communications.

6.1.2. Stateless Configuration with DHCPv6

Sometimes, DHCPv6 will be used alongside SLAAC. SLAAC will provide the address assignment, and DHCPv6 will provide additional host configuration options, such as DNS servers. If any of the DHCPv6 options are directly related to the IPv6 addresses being renumbered, then the configuration must be changed at the appropriate time during the renumbering event, even though it itself does not handle the address assignments.

Since the configuration is stateless, the DHCPv6 server will not know which clients to contact to inform them to refresh. Clients of the configuration protocol should poll the service to obtain potentially updated ancillary data, such as suggested by [29]. It is proposed that a new DHCPv6 service option is added to inform clients of an upper bound for how long they should wait before re-requesting service information.

6.1.3. Tokenised Interface Identifiers

IPv6 Stateless Address Auto-configuration (SLAAC) enables network administrators to deploy devices in a network and have those devices automatically generate global addresses without any administrative intervention, and without the need for any stateful configuration service such as DHCPv6.

However, certain services - such as HTTP, SMTP and IMAP - may better benefit from having 'well known' identifiers that do not depend on the physical hardware address of the server's network interface card, e.g. <prefix>::53 for name servers.

Tokenised addresses offer a facility for administrators to specify

the bottom 64 bits of an IPv6 address for a node whilst allowing the top 64 bits (the network prefix) to be automatically configured from router advertisements.

Currently, only more recent versions of Sun Microsytems' Solaris operating system features ioctl-configured support for tokenised interface identifiers, although recent work at Southampton has demonstrated that the configuration technique can be introduced trivially through simple kernel extensions in Linux.

As regards renumbering, automatically configured tokenised addresses, where the network prefix component is learnt through router advertisements, ease the renumbering process where administrators have elected to use well known interface identifiers. Rather than having to manually reconfigure the nodes with the new addresses, the nodes can rely on automatic configuration techniques to pick up the new prefix.

6.2. Stateful Configuration with DHCPv6

As opposed to stateless autoconfiguration, IPv6 stateful or managed configuration can be achieved through the deployment of DHCPv6. Section 18.1.8 of [30] details how a node should respond to the receipt of stateful configuration data from a DHCPv6 server where the lifetime indicated has expired (is zero). Section 19.4.1 details how clients should respond to being instructed by DHCPv6 servers to reconfigure (potentially forceful renumbering). Section 22.6 details how prefix validity time is conveyed (c.f. the equivalent data in SLAAC's Router Advertisement).

In order to renumber such a network, the DHCPv6 server should send reconfigure messages to inform the clients that the configuration has changed, and the clients should re-request configuration details from the DHCPv6 server. This, of course, relies on the clients correctly responding to such messages.

Where DHCPv6 has been employed, careful consideration about the configuration of the service is required such that administrators can be confident that clients will re-contact the service to refresh their configuration data. As alluded to in sections 22.4 and 22.5 of [30], the configurable timers that offer servers the ability to control when clients re-contacts the server about its configuration can be set such that clients rarely (if ever) connect to validate their configuration set.

The approach described in [29] allows the lifetime of other configuration information supplied by DHCPv6 to be ramped down in preparation for a planned renumbering event.

6.2.1. Prefix Delegation

Where stateless autoconfiguration enables hosts to request prefixes from link-attached routers, prefix delegation enables routers to request a prefix for advertising from superior routers, i.e. routers closer to the top of the prefix hierarchy - typically topologically closer, therefore, to the provider. Once the router has been delegated prefix(es), it can begin advertising it to the connected subnet (perhaps even multi-link) with indicators for hosts to use stateful (DHCPv6) or stateless address autoconfiguration as per RFC2461.

There have been two principal approaches to prefix delegation proposed: HPD (Hierarchical Prefix Delegation for IPv6), which proposed the use of bespoke ICMPv6 messages for prefix delegation, and IPv6 Prefix Options for Dynamic Host Configuration Protocol [31], which defines a DHCPv6 option type. Of the two approaches, the DHCPv6-based approach has received wide support and is on the standards track.

6.2.2. Source Address Selection Policy distribution

It has been proposed that DHCPv6 could also be used to distribute source address selection policy to nodes [24]. The model proposes that consumer edge router receives policies (e.g. from multiple ISPs in the case of multi-homed networks) and re-distributes them to end nodes. The end nodes then put them into their local policy table, which leads to appropriate source address selection. Where the design goal was a distribution mechanism in light of multi-homed networks, the adoption of the technique for the multi-prefix states of [1] during renumbering appears appropriate.

6.3. Router Renumbering

RFC2894 [7] defines a mechanism for renumbering IPv6 routers throughout a network using a bespoke ICMP message type for manipulating the set of prefixes deployed throughout subnets. Through the use of prefix matching and a rudimentary algebra for bit-wise manipulation of prefix data bound to router interfaces, the mechanism enables administrators to affect every router within a scope from a single administration workstation. One drawback of RFC2894 is that it requires an enterprise-wide IPsec infrastructure to be deployed to secure the ICMP messages in order to be compliant.

The approach utilises multicast communication to the all-routers address, FF05::2, scoped to the entire 'site' as determined by router filter policy to distribute configuration updates to all (compliant) routers. The mechanism also works with more specific addressing

modalities, such as link-local multicast (FF02::2) to reach all routers on a specific link, or directed unicast to affect a specific router instance. When surveying current implementations very few IPv6 implementations bound their interfaces to the Site-wide All-Routers multicast address (FF05::2), and fewer still have implementations of RFC2894.

Example use cases cited in RFC2894 are for deploying global routing prefixes across a hierarchical network where site-locals already exist (presumably updated now to Unique Local Addresses), and for renumbering from an existing prefix to another in a similar manner to that proposed by Baker (i.e. the deployment of a new prefix alongside the existing one, which is deprecated and subsequently expired and removed - using the same mechanism described).

The specification was developed before the shift in recommendation away from the Top-, Next- at Site-Level Aggregation Identifier address allocation hierarchy of RFC3513, although the techniques documented for renumbering the global routing prefix and subnet ID components in the updated address allocation recommendations [17] are not affected by the architectural change.

As with other prefix assignment techniques, it is the responsibility of the node to correctly deprecate and then expire the use of a previously assigned prefix as defined by the IPv6 Neighbour Discovery protocol, RFC2461 [8], section 4.6.2 describing the Prefix Information option in particular.

7. Administrative Considerations for Renumbering

This section is concerned with factors that affect the renumbering procedure, from a network administration viewpoint. In particular, this section discusses areas that a network administrator should consider before undertaking a renumbering event, to ensure that it proceeds smoothly. This includes considerations of event frequency, scalability, and those relating to delays in information propagation.

7.1. Router Advertisement Lifetimes

As discussed in Section 6.1.1, IPv6 Stateless Autoconfiguration allows the expiration of assigned prefixes. This process permits existing sessions to continue while preferring a new prefix. It should be noted, however, that there are some limitations in the specification that have an impact in renumbering. In particular, it is not possible to reduce a prefix's lifetime to below two hours if it has previously been available at a longer validity. This therefore emphasises the need to plan renumbering events in advance

if at all possible, to reduce the lifetime as required, within these limitations.

7.2. Border filtering

Multi-addressing (Section 5.1) allows multiple globally reachable addresses to be assigned to node interfaces, but one administrative caveat that arises is that of site border filtering. Not only is it the norm for sites to filter at their border router traffic that is not destined to local subnets, but it is also increasingly common for sites to undertake egress filtering. This is often used to prevent administratively local addresses (such as the, now deprecated, site-local prefix) 'leaking' traffic, or for mis-configured hosts (e.g. visitors with manually configured stacks without Mobile IPv6) from sourcing traffic that cannot be routed back (cases of which may include deliberate IP spoofing or DDoS attempts).

Providers often use ingress filtering so that the provider only accepts packets from customers that have source addresses inside the address space the provider has delegated to the customer. With multi-addressing, hosts in the site may send packets with source addresses from either provider's address space. If the providers do ingress filtering, a packet must then be forwarded out on the correct uplink, based on which source address the packet has. If the site has a common exit router for the two uplinks, that router will need to route the packets based on the source address. If the site has two different exit routers, the entire site backbone may need to route based on source addresses in order to forward the packets to the correct exit router.

7.3. Frequency of renumbering episodes

The many different renumbering scenarios, discussed in Section 3, can have vastly different frequencies of renumbering events. In the case of a provider offering only dynamically assigned IP addresses, it could be very frequent, for example as frequent as 'per-connection' for dial-on-demand services, or weekly for some broadband services. Such renumbering events usually only occur when a customer reconnects to such services or are explicitly cited in a subscription agreement and as such are often pre-determined.

The renumbering of a site due to upstream renumbering is relevant to all connections from a small dial-up link to a large enterprise. It is of particular interest since the end user has no control over the timing or frequency of the renumbering events. It is expected, however, that such events are likely to be very infrequent.

The other irregular renumbering events are those that occur due to

end user migrating, either to a new provider, or to a new address allocation of their choosing. The timing of such an event is therefore often within the control of the end user (within reason), and are also likely to be one-off events, or at the very least, highly infrequent.

7.4. Delay-related Considerations

When considering a renumbering event, both the planning of, and responses to the event are affected by temporal factors. The amount of time available in which to undertake the operation can change the administrative actions required, and this section aims to discuss some of these issues.

7.4.1. With or without a flag day

A network may be renumbered with or without a flag day. In the context of this document we are focusing on without a flag day, although many of the issues will still apply when renumbering is effected with a flag day.

Despite the similarities, because there is an outage of services when renumbering with a flag day, it is not necessary to ensure continuity of network connections, and almost all reconfiguration can be done during the outage, thus greatly simplifying the task of renumbering.

7.4.2. Freshness of service data

One of the largest issues when renumbering a network will be the effect on applications that are already running. In particular, applications that periodically contact a particular host may do an initial hostname lookup, and cache the result for use throughout the lifetime of the program. In such a situation, there is no way for the application to find out that the host in question has been renumbered, and it should stop using its already cached address. It is therefore recommended that applications should regularly request hostname lookups for the desired hosts, leaving the caching to the resolver. It is then up to the resolver to ensure that resource record TTLs are observed, and its cached response is updated as necessary.

Despite this, there is still a serious issue in that there is no method of caching resolvers knowing when a renumbering event is going to take place. If a typical RR's TTL is one day, then that should be reduced not less than a day before the renumbering event, so that resolvers will more frequently check for changed records. This will work successfully for a pre-planned renumbering event, but problems of stale, cached records will exist if the renumbering event is

unplanned (e.g. by receiving a new router advertisement from upstream).

There are also cases where the use of a resolver is not practical, such as with packet filter rules. If a packet filter has been configured with explicit hostnames, these are translated to IP addresses for fast packet matching. The per-packet resolver function is highly undesirable from a pure performance perspective. Such a packet filter is likely to need to be reloaded for the DNS changes to be recognised.

A similar problem exists when a nameserver is renumbered. If the operating system's resolver has cached the nameserver address, it will at some point find it unavailable. To mitigate this problem, it is suggested that at least one off-site nameserver is included in the configuration. In addition, well-known anycast addresses (see Section 5.6) could be used, so that the client's DNS configuration does not need to be changed at all during the renumbering event.

The basic process of renumbering, involving the introduction of a new prefix and the deprecation and eventual removal of the old prefix, could be hypothetically handled by a special tool, with no manual intervention. Such a tool would have to become significantly more complex in order to handle all the cases where IP addresses are explicitly specified (a comprehensive list is given in Section 9.2). Other particularly notable cases that could be changed with a tool, were it to be developed, include DNS zone files and DHCPv6 configuration. Deployment of such a tool, even if possible, would be made complex through the requirement to authenticate the updates to each instance of the deployed literals.

7.4.3. Availability of old prefix

The duration of the period where the old prefix remains available affects the length of time that can be allowed for the renumbering procedure, and the maximum time for which existing sessions could continue. If end users have control over the renumbering procedure (such as when changing provider), then they can continue providing the old prefix for as long as required, within reason (such as cost aspects). This heavily mitigates the issues of session survivability, and relaxes the speed at which hosts must be reconfigured.

If the end users do not have such control, such as when the upstream provider forces the renumbering, the availability of the old prefix is determined entirely by the upstream provider's willingness to continue providing it, which is likely to be based on the technicalities of their own renumbering situation. The end user

should therefore not rely on retaining the old prefix for a relatively long period of time. In addition, many situations, such as dial-on-demand with dynamic IP addresses, and nomadic networks, will lose their old prefix quickly, if not almost instantaneously.

It would be possible to continue using the old prefix internally, even when the external connectivity for that prefix is no longer active, for example to keep access to core services such as DNS servers while the transition is taking place. This should, however, be considered bad practice in case of route leaking and application confusion, as well as preventing access to the addresses if they have been reassigned, and as such this should only be used as a last resort to ensure internal continuity of service, if the availability of the old prefix is too short to allow a full transition to take place.

7.4.4. Duration of overlap

A key operational decision when renumbering is enforced due to a change in connectivity provider is how long to sustain the overlap of two live prefixes. The trade-off to be made is the cost of maintaining two contracts with separate providers against the 'smoothness' of the transition to the new prefix as regards local administration overheads, service migration, etc. Where larger corporations can likely suffer the increased financial costs, SMEs and SOHOs might consider as little as one month's overlap too expensive, and so Baker's State 5 (Stable use of either prefix) [1] is unattainable in such scenarios.

In some cases, there may be technical reasons for the overlap to not be feasible, such as with xDSL provision where the new service is a drop-in replacement for the old and the two cannot co-exist (for example, because the provision of the service requires the whole circuit resource from exchange to customer).

7.5. Scalability issues

During the renumbering transition, there will be a time when two prefixes are valid for use. At this point, there will be a considerable amount of configuration that will have to be (temporarily) duplicated. In particular, routing entries on the hosts will be doubled, and there will, for a short period, be two forward DNS records for every hostname. Security is another key scalability issue. All access control lists, packet filters, etc, will need to be updated to cope with the multiple addresses that each host will have. This could have a noticeable impact on packet filter performance, especially if it lead to, for example, the doubling of several hundred firewall rules.

The scalability issues created by the increase in configuration to cope with the temporary existence of multiple addresses per host adds a complexity in management, but how much so is up to the end-users themselves. A user may choose to do direct transitions of some services (such as web servers) from one IP address to another, without going through a stage where the service is available on all addresses. While that is not strictly providing a fully seamless transition, it could significantly reduce the management complexity, without a significant impact on service, especially if the DNS updates are rapid.

It should also be noted that during a renumbering event, since the DNS resource record TTLs are significantly shorter, the primary DNS servers for the domains will receive significantly more queries, as resolvers should not cache the responses for so long, and will regularly check back with the master. The likelihood of this having any significant impact is, however, fairly minimal, at least in a typical small to medium site.

Section 3.1 of Baker [1] is aptly titled "Find all the places", and serves as a gentle reminder to application developers that embedding addresses is bad at best. Where common UNIX tools such as "grep" allow administrators to crawl the file systems of servers for places where address information is hard-coded, the proliferation of technologies such as NetInfo and other directory- or hive-based configuration schemes makes the job of finding all the places that addresses are hard-coded intractable.

Beyond the call to arms for application and services developers made by Baker et al. [1], and specific to the challenges of renumbering, the following security and policy-related services that initial research has flagged as particularly troublesome:

7.5.1. Packet filters, Firewalls and ACLs

Throughout the transition from the old address set to the new, all packet filters and firewalls will need to adapt to map policy to both prefixes (sets of addresses) - perhaps even selectively as the old addresses become deprecated. Whilst technologies such as Router Renumbering and Neighbour Discovery automate to a large extent the transition of router and node configurations, and dynamic DNS update for the re-mapping of resource records to reflect the new addresses [32], no such mechanism exists at present for mechanising the adaption of security policy.

Particularly troublesome policies to administer include egress filtering, where packet filters discard outbound packets that have source addresses that should not exist within the site, and filtering

inbound site-local addresses in cases where two organisations are renumbering as a step toward merging their networks together (although the use of site-local addressing is now deprecated).

Where renumbering is due to a 'clean break' from previous connectivity provider, another consideration is for the ingress filtering performed by the provider. For instance, the new provider may refuse to receive into their routing topology those packets whose source address is under the old prefix, and likewise for the old provider and new prefix. Whilst it is not the business of the IETF to mandate business practice, it is likely that the provision of out-of-allocation prefix routing as part of a multi-homing service contract would be a chargeable service and not one that an enterprise trying to make a clean break away would likely be willing to pay just for the duration of transition to their new prefix.

Beyond the immediate up-stream provider, there are other policy-based considerations to take into account when renumbering. Some rudimentary authenticated access mechanisms rely on access queries coming from a particular IP network, for example, and so those application service providers will need to update their access control lists. Likewise all the internal applications (possibly meant for 'internal' eyes only) will have to have their access controls updated to reflect the change. The use of symbolic access controls (i.e. DNS domain names) rather than embedded addresses may serve to mitigate much of the distributed administrative load here, at least if such symbols are re-resolved, especially during the mid-renumbering states where both sets of addresses are still live and valid.

7.5.1.1. Policy rule replication where both prefixes valid

One key caveat with policy application during a renumbering prefix concerns rules that are 'tied down' at both ends to (sets of) addresses under the prefix to be renumbered, i.e. those that detail specific nodes or subnets in both source and destination elements of the policy rule as opposed to source 'any' or destination 'any'.

Examples of where this approach apply include specific holes punched through a packet filter between a DMZ and the internal network, e.g. for staged access to compute servers from off-site.

A dilemma here is that the otherwise 'ideal practice' use of symbolic names to identify elements in the network may not be appropriate in policy rules. This is particularly the case where resolver libraries do not return all bound resource data for symbols (i.e. old and new AAAA records for www.example.com), or where policy applications do not iterate across all returned resource record data in resolvers

that are well behaved. It also assumes that name service data is updated ahead of policy application, which is ill-advised given that the instant name servers start serving data regarding new, yet to be configured, addresses for nodes.

7.5.2. Monitoring tools

Network monitoring and supervisory utilities such as RMON probes, etc., are often deployed to monitor network status based on IP traffic. During a renumbering episode, the addresses for which the probes should monitoring and the addresses of logging services to which the probes report (e.g. in the case of remote SNMP logging) need to be tracked.

"Helpdesk ops" service liveness monitoring software also poses a particular problem where liveness is determined, for example, by a null transaction (e.g. for POP3 mail server, authenticating and performing a NOOP) made against a named service instance, if the name is by IP then two instances of the liveness test will be required: one on the old address to cater for those remote parties that are not yet aware of the new address, and one test against the new.

As part of the renumbering process, it may be advantageous to deploy flow analysis tools that can be scripted to alert administrators on observation of particular traffic patterns, e.g. flows to a service under a deprecated prefix during transitions where both old and new prefixes are live and routed to the site concurrently. This can highlight, for example, mis-cached DNS resource records, sources of manually configured service location data, etc.

When relying on DNS labels for identifying nodes to administer, care must be taken to ensure that the complete set of nodes administered are caught. For instance, a set of application servers may share the same DNS label and rely on DNS round-robin for rudimentary load balancing (a modality at odds with the notion of maintaining resource records for both old and new prefixes during renumbering episodes). A network monitoring tool that was configured to monitor just that service that was resolved by address lookup might only capture one of that set of nodes.

7.6. Considerations with a Dual-Stack Network

There are several issues to consider when renumbering a dual-stacked network. In the simplest case, the IPv4 addresses will be remaining the same while the IPv6 addresses are renumbered. This could, for example, be due to an upstream renumbering, a change of IPv6 transition method (such as a tunnel), or a topology change. In such a case, the IPv4 connectivity remains unchanged, and as such can be

used as a fallback during the renumbering to assist with session continuity, DNS services, etc.

The other case is when the IPv4 network is being renumbered along with the IPv6 network. Again this could be due to an upstream change, a network reconfiguration, or because the two are inter-linked - such as with the 6to4 transition mechanism. In this case, it is unlikely that the existence of IPv4 on the network can be used for any advantage, and instead many of the same issues are likely to be found when renumbering the IPv4 network as for the IPv6 network, except for the fact that more of the renumbering must be manually configured, for example by reconfiguring the stateful IPv4 DHCP configuration, or even manually configuring IPv4 addresses.

A hybrid case is also possible, where IPv4 NAT is used on the internal network, but with globally routable IPv6 addresses. In this case, if both networks' external connectivity is being renumbered, the internal network will only see the effect of the IPv6 renumbering, while keeping the IPv4 addresses the same. The renumbering procedure will still have an impact on the IPv4 connectivity and its session survivability, however. It may also be possible that the site uses both global and ULA IPv6 prefixes, the ULA prefix being deployed to avoid impact to long-running IPv6 sessions.

7.7. Equipment administrative ownership

The question of who owns and administers (also, who is authorised to administer) the site's access router is an issue in some renumbering situations. In the enterprise scenarios, the liaison between the end users and remote administrators is likely to be relatively easy; this is less likely to be the case for a SOHO scenario. This is not likely to be a major issue, however, since SOHO renumbering is likely to only be required if the remote administrators deem it necessary, or if the end user is sufficiently technically competent and decides to renumber their own network.

8. Impact of Topology Design on Renumbering

This section looks at considerations regarding network design, such as network merging, and design-time recommendations that can help avoid the need for a network renumbering event.

8.1. Merging networks

Renumbering of all or part of a network due to merging two or more smaller networks has many of the concerns already discussed, but it

may not affect the whole network. For example, multiple disparate networks may be merged together as one entirely new subnet, and thus all hosts must be renumbered; but it is also possible that one of the networks in the merger retains its prefix, and the other network(s) merge with it.

When the networks merge, the router advertises itself, and the new prefix if appropriate, to the new hosts, and Duplicate Address Detection (DAD, see Section 5.4 of [6]) must be applied by the new hosts to ensure they are not taking addresses already assigned to the existing hosts. It is implementation-dependent, however, as to whether the DAD algorithm will be re-run on link-local addresses if the network configuration is changed, so there is the possibility of an address conflict. However, as is noted in RFC2462, DAD is not completely reliable, and as such it cannot be assumed that initially after a network merge all link-local addresses will be unique.

8.2. Fixed length subnets

The IAB/IESG recommendations for IPv6 address allocations [10] details some of the motivations behind the change in the addressing architecture of IPv6 since its inception, and asserts the current state of a 64-bit 'network' part (the prefix) and a 64-bit 'host' part (the interface identifier). Fixing the lower 64 bits to be exclusive of routing topology significantly reduces the administrative load associated with renumbering and re-subnetting as experienced with IPv4 networks previously, for example, to get better address utilisation efficiency as networks evolve and provider address allocations changed.

The recommendations also discuss what length of network prefix should be allocated to sites, typically provisioning for 16-bits of subnet space in which sites can build their topology. Having such a large address space for sites to divide up at their discretion alleviates many of the drivers for renumbering discussed during the PIER working group's lifetime [3].

8.3. Use 112-bit prefixes for point-to-point links

It is recommended that point-to-point links, such as tunnel endpoints or router-router links, are allocated /112 subnets from a single /64 within the site's allocation. This simplifies policy-based filtering and is less wasteful of address space than using /64s everywhere, improving the address utilisation ratio for the site that would in extreme cases lead to a larger prefix becoming required.

The 112-bit prefix length is preferred to 127-bit on the advice of RFC3627[33], which suggests that such allocations can lead to end-

point address starvation where one router elects to take both the zeroth address in the /127 as a subnet router anycast address and the first address for its endpoint, leaving no address for the remote end of the link.

8.4. Plan for growth where possible

When designing address topology - particularly in ISP and larger-scale Enterprise sites - it is recommended that network designers plan for growth of lower hierarchies under their provision (e.g. a /60 satellite site becoming big enough for a /56; a /48 customer getting sufficiently large as to warrant a shorter prefix).

Techniques for such allocations include centre-most bitset growth as described in Section 3.3 of RFC3531 [17], which leave the bits nearer upstream and downstream bit-boundaries until much later in the allocation selection set, meaning that a boundary shift has minimal impact on existing deployed allocations. However the overheads and non-contiguous nature of successive allocations may not suit Enterprise sites, meaning that other allocation strategies are required, contextually sensitive to the demands of the site in which the prefixes are being deployed.

In enterprise networks where satellite sites participate, it is recommended that single-subnet blocks are skipped in the allocation such that remote satellites can grow (double) without requiring those `nearby' in the address block to renumber.

For example, the strategy taken in an enterprise with 56-bit prefixes allocated to satellites is to leave subsequent /56s for future expansion of each sub-tier to a /55.

Note that strictly adopting RFC3531 may be insufficient in enterprises where, for example, there is a mix of subnet provision (e.g. for satellite sites) and end-user subnets.

8.5. IPv6 NAT Avoidance

RFC2072 stated: "Network address translation (NAT) is a valuable technique for renumbering, or even for avoiding the need to renumber significant parts of an enterprise." That is, by 'hiding' the subnet topology and making independent of any connectivity provider the addressing model used within a site, NATs enable renumbering of entire networks because the only device that is renumbered when global addressing changes is the outside edge of the NAT devices.

However, NAT is strongly discouraged in IPv6, not least because it breaks end-to-end transparency (as described in [34]) and obscures

identity - including the basis for permission, authorisation, verification and validation - and thus should not be considered as being available as a solution. A significant reason to deploy IPv6 is to simplify network and application operation by (IPv4) NAT removal, for example to provide true end-to-end connectivity, to make simple the gateway between site and Internet, to encourage 'considered' policy for secure access rather than rely on the (relatively) dangerous defence of 'hiding' behind a NAT. A more detailed discussion of the motivations for 'protecting' the network architecture from NATs can be found in [35].

9. Application and service-oriented Issues

In this section we highlight issues and common approaches to software development that 'disrupt' protocol layering to the extent that applications become aware of renumbering episodes, even if catastrophic and without knowing how to recover without failing.

NOTE: This section, like the discussion sections before it, will evolve as experience grows researching the various renumbering strategies in controlled experiments - particularly in light of Section 10.1.

9.1. Shims and sockets

As discussed in Section 7.5, Baker's draft calls for application developers to consider the effects of renumbering whilst applications are 'live', particularly as regards caching the results of symbol resolution. Where applications maintain open connections to services over a sustained period of time (as opposed to the ephemeral nature of protocol interactions such as with HTTP), any change in either end's addressing may intrude on the application's execution - particularly if the change is abrupt or the session longer than the expiry and withdrawal time of the old addresses.

Various options may be available to minimise the risk of application disruption in this instance. A HIP-like 'shim' [36], as is being developed as a candidate solution to the general multi-homing problem, removes the tight coupling between a connection and a service's topological location: as the renumbering event takes place, the locator is updated to reflect the new address topology, and the application remains blissfully unaware - a form of layer 3.5 mobility.

Alternatively, should the old address space be available such that a single (or subnet of) Mobile IPv6 Home Agents be deployed in the routing path of the to-be-otherwise-interrupted connection, then the

endpoint being renumbered could utilise layer 3 mobility once the old prefix is removed from its link, i.e. register with the Home Agent in the old prefix topology - presumably in the provider's network, formerly upstream from the site - and rely on Mobile IPv6 route optimisation to make good the additional overhead imposed by the reverse tunnelling to the new prefix.

Applications that employ SCTP as opposed to TCP or UDP for communication avoid all of the issues highlighted in this sub-section due to the provision of dynamic endpoint reconfiguration in the protocol (see Section 4.2).

9.2. Explicitly named IP addresses

There are many places in the network where IP addresses are embedded as opposed to symbolic names, and finding them all to be updated during a renumbering episode is not a trivial task. This section details an evolving list of such places as surveyed as common.

Addresses may be hard-coded in software configuration files or services, in software source-code itself (which is particularly cumbersome if no source is available, e.g. a bespoke utility built to order), in firmware (for example, an access-controlling hardware dongle), or even in hardware, e.g. fixed by DIP switches.

A non-exhaustive list of instances of such addresses includes:

- Provider based prefix(es)
- Names resolved to IP addresses in firewall at startup time
- IP addresses in remote firewalls allowing access to remote services
- IP-based authentication in remote systems allowing access to online bibliographic resources
- IP address of both tunnel end points for IPv6 in IPv4 tunnel
- Hard-coded IP subnet configuration information
- IP addresses for static route targets
- Blocked SMTP server IP list (spam sources)
- Web .htaccess and remote access controls

- Apache .Listen. directive on given IP address
- Configured multicast rendezvous point
- TCP wrapper files
- Samba configuration files
- DNS resolv.conf on Unix
- Any network traffic monitoring tool
- NIS/ypbind via the hosts file
- Some interface configurations
- Unix portmap security masks
- NIS security masks
- PIM-SM Rendezvous Point address on multicast routers

Some hard-coded IP address information will be held in remote locations, e.g. remote firewalls, DNS glue, etc. adding to the complexity of the search for all instances of the old prefix. Should symbols be used rather than addresses, administrative ownership of DNS - with due consideration for the TTL of resource records - and other naming services ease this particularly problematic issue of data ownership and validity.

There are also cases when IP addresses are embedded into payload data, such as with UDP-based NFS mounts and FTP sessions. These cases were discussed in more detail in Section 4.2.4.

9.3. API dilemma

In light of Section 7.4.2, there is an open question as to whether we need an extension to the sockets API that would allow applications resolving addresses to be able to determine the freshness of the resolved data. A straw poll of networking applications demonstrated that common programming practise is to 'resolve once, bind many' during the lifetime of an application, caching the initial lookup result and assuming that it is still valid throughout. Whilst this is a perfectly valid approach for short-lived applications, where the chance of renumbering - site or the single node - increases with regards the longevity of the application, the likelihood of the resolved data being intrusively inaccurate also increases.

Application programmers should therefore consider the possibility of network renumbering when writing socket software. The best behaviour is probably to freshly resolve for any socket binding, and let the resolver handle the caching, based on the DNS TTL. Only when there are a significant number of connections within a short timeframe should application-level caching be considered.

9.4. Server Sockets

Certain applications create a server socket and bind the socket so that they only receive connections or datagrams at one specific address. These services typically keep the socket bound to that address until they are shut-down or restarted. This means that if the host is configured with a new address, these applications would not respond to that address.

If the applications were listening to the wildcard address, they would also accept connections and datagrams on new addresses as they become configured on a node.

An example would be a webserver, which may in fact bind to multiple different IP addresses to serve content for different domains where the particular business case is for customers to be allocated their 'own' IP address (e.g. for reverse DNS to reflect their branded domain name).

A typical work-around would be to schedule a restart of all such services having first identified whether they can operate on both address prefixes (to satisfy the middle states of Baker [1]), or at least to schedule their migration to the new address configuration in light of the DNS name bindings (considering caches and TTL), and the nature of existing clients that may still be bound to the old service (consider graceful migration).

One possible solution, not implemented in existing socket APIs, would be to allow servers to bind to just the lowest 64 bits of an address, allowing the network identifier to change without the server knowing. This is a purely hypothetical solution, however, and has numerous issues, not least regarding requirements of some server software to know its current globally routable IP address.

9.5. Sockets surviving invalidity

When an address expires (validity lifetime falls to zero), addresses are to be removed from interfaces, and the expired address is not to be used as a source address for further packets (see RFC2462 section 5.5.4 and RFC2215 secion 10).

However, it appears that for an established TCP session or for UDP where the application has bound to a specific address, many stack implementations keep using the same source address blindly putting packets onto the wire, even if the address is removed from the interface.

It appears that these stack implementations make sure the address is valid when the TCP session is created or when an application binds to an address on a datagram socket, but once the socket is bound to that address there are no more checks.

Whilst this is not a serious issue - certainly, no reply packets could be received as the interface will not listen for them, and it is likely that the prefix would no longer be routable at the next-hop router beyond the point of invalidation - it does mean that application data will be lost up until that point where the transport layer determines that the packets are not being received (e.g. TCP ACKs).

9.6. DNS Authority

It is often the case in enterprises that host web servers and application servers on behalf of collaborators and customers that DNS zones out of the administrative control of the host maintain resource records concerning addresses for nodes out of their control.

The upshot here is that when the service host renumbers, they do not have sufficient authority to change the AAAA records, etc., that refer to newly renumbered addresses.

It is recommended that remote DNSes maintain CNAME records to labels in a zone that is under the authoritative control of the enterprise whose addresses are referenced.

10. Summary

This memo has further motivated the issue of network renumbering, highlighted important requirements to ensure that episodes can pass smoothly with a minimum of disruption to users, and indicated a number of protocol features and technologies that assist network designers and operators in the smooth transition from one prefix to another, all in the context of [1].

10.1. IETF Call to Arms

Validation surveys of address selection implementations per RFC3484, of address expiry per RFC2462 and RFC3315, and operational experience

validating the Baker et al. procedure have been carried out and reported on in other fora (e.g. in D3.6.1 of the 6NET project). However, in the above considerations, a number of actions would be most helpful in advancing the understanding of the practical implications and robustness of IPv6 renumbering. These include:

- Survey of the pervasiveness of address literals and steps to avoid their use
- Validation of address selection at source and destination during various stages of Baker's renumbering procedure in implementations other than Cisco IOS, FreeBSD 5.9, Linux 2.6, Macintosh OS/X 10.4, Sun Solaris 8-10, Microsoft Windows XP SP2
- Validation of RA lifetime expiry and confirmation of prefix removal and effects on existing sessions in other implementations
- Validation of IPv6 Prefix Delegation by DHCP, and of IPv6 Router Renumbering
- Better understanding of the commonalities and differences between renumbering and multi-homing
- Anecdotal experience of IETF members that have undertaken an IPv6 renumbering exercise, e.g. in the transition from 3FFE::/16 6Bone addresses to production GAU

Given that this memo is dressed as a set of "things to think about", there is no conclusion other than a call for input from the IETF community.

There may be a case to be made to reopen the PIER WG in the new context of IPv6, although that group has not been active since 1997.

11. IANA Considerations

This document makes no request of IANA.

12. Security Considerations

The security considerations as outlined in [1] still hold, with the following supporting comments... (tbd)

13. Acknowledgements

The authors gratefully acknowledge the many helpful discussions and suggestions of their colleagues from the 6NET consortium, particularly Fred Baker, Graca Carvalho, Ralph Droms, David Mills, Thorsten Kuefer, Eliot Lear, Christian Schild, Andre Stolze, Tina Strauf, Bernard Tuy, and Gunter Van de Velde.

14. References

14.1. Normative References

[1] Baker, F., "Procedures for Renumbering an IPv6 Network without a Flag Day", draft-ietf-v6ops-renumbering-procedure-05 (work in progress), March 2005.

[2] Berkowitz, H., Ferguson, P., Leland, W., and P. Nesser, "Enterprise Renumbering: Experience and Information Solicitation", RFC 1916, February 1996.

[3] Ferguson, P. and H. Berkowitz, "Network Renumbering Overview: Why would I want it and what is it anyway?", RFC 2071, January 1997.

[4] Berkowitz, H., "Router Renumbering Guide", RFC 2072, January 1997.

[5] Draves, R., "Default Address Selection for Internet Protocol version 6 (IPv6)", RFC 3484, February 2003.

[6] Thomson, S. and T. Narten, "IPv6 Stateless Address Autoconfiguration", RFC 2462, December 1998.

[7] Crawford, M., "Router Renumbering for IPv6", RFC 2894, August 2000.

[8] Narten, T., Nordmark, E., and W. Simpson, "Neighbor Discovery for IP Version 6 (IPv6)", RFC 2461, December 1998.

14.2. Informative References

[9] Carpenter, B. and K. Moore, "Connection of IPv6 Domains via IPv4 Clouds", RFC 3056, February 2001.

[10] IAB and IESG, "IAB/IESG Recommendations on IPv6 Address Allocations to Sites", RFC 3177, September 2001.

[11] Fink, R. and R. Hinden, "6bone (IPv6 Testing Address Allocation) Phaseout", RFC 3701, March 2004.

[12] Templin, F., Gleeson, T., Talwar, M., and D. Thaler, "Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)", draft-ietf-ngtrans-isatap-24 (work in progress), January 2005.

[13] Narten, T. and R. Draves, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 3041, January 2001.

[14] Ernst, T. and H. Lach, "Network Mobility Support Terminology", draft-ietf-nemo-terminology-05 (work in progress), March 2006.

[15] Stewart, R., Xie, Q., Morneault, K., Sharp, C., Schwarzbauer, H., Taylor, T., Rytina, I., Kalla, M., Zhang, L., and V. Paxson, "Stream Control Transmission Protocol", RFC 2960, October 2000.

[16] Stewart, R., "Stream Control Transmission Protocol (SCTP) Dynamic Address Reconfiguration", draft-ietf-tsvwg-addip-sctp-15 (work in progress), June 2006.

[17] Hinden, R. and S. Deering, "Internet Protocol Version 6 (IPv6) Addressing Architecture", RFC 3513, April 2003.

[18] Johnson, D., Perkins, C., and J. Arkko, "Mobility Support in IPv6", RFC 3775, June 2004.

[19] Huston, G., "Architectural Approaches to Multi-Homing for IPv6", draft-ietf-multi6-architecture-04 (work in progress), February 2005.

[20] Haberman, B. and D. Thaler, "Unicast-Prefix-based IPv6 Multicast Addresses", RFC 3306, August 2002.

[21] Savola, P. and B. Haberman, "Embedding the Rendezvous Point (RP) Address in an IPv6 Multicast Address", RFC 3956, November 2004.

[22] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", draft-ietf-ipv6-unique-local-addr-09 (work in progress), January 2005.

[23] Hinden, R. and B. Haberman, "Centrally Assigned Unique Local IPv6 Unicast Addresses", draft-ietf-ipv6-ula-central-01 (work in progress), February 2005.

[24] Matsumoto, A., "Source Address Selection Policy Distribution for Multihoming", draft-arifumi-multi6-sas-policy-dist-00 (work in progress), October 2004.

[25] Johnson, D. and S. Deering, "Reserved IPv6 Subnet Anycast Addresses", RFC 2526, March 1999.

[26] Jeong, J., "IPv6 Host Configuration of DNS Server Information Approaches", draft-ietf-dnsop-ipv6-dns-configuration-06 (work in progress), May 2005.

[27] Abley, J. and K. Lindqvist, "Operation of Anycast Services", draft-ietf-grow-anycast-04 (work in progress), July 2006.

[28] Partridge, C., Mendez, T., and W. Milliken, "Host Anycasting Service", RFC 1546, November 1993.

[29] Venaas, S. and T. Chown, "Information Refresh Time Option for DHCPv6", draft-ietf-dhc-lifetime-03 (work in progress), January 2005.

[30] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[31] Troan, O. and R. Droms, "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6", RFC 3633, December 2003.

[32] Vixie, P., Thomson, S., Rekhter, Y., and J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", RFC 2136, April 1997.

[33] Savola, P., "Use of /127 Prefix Length Between Routers Considered Harmful", RFC 3627, September 2003.

[34] Carpenter, B., "Internet Transparency", RFC 2775, February 2000.

[35] Velde, G., "IPv6 Network Architecture Protection", draft-ietf-v6ops-nap-03 (work in progress), July 2006.

[36] Moskowitz, R. and P. Nikander, "Host Identity Protocol Architecture", draft-ietf-hip-arch-03 (work in progress), August 2005.

URIs

[38] <http://www.ietf.org/html.charters/multi6-charter.html>

[39] <http://www.ietf.org/html.charters/shim6-charter.html>

Authors' Addresses

Tim J. Chown
University of Southampton, UK
Electronics and Computer Science
University of Southampton
Southampton SO17 1BJ
UK

Phone: +44 23 8059 5415
Fax: +44 23 8059 2865
Email: tjc@ecs.soton.ac.uk

Mark K. Thompson
University of Southampton, UK

Email: mkt@ecs.soton.ac.uk

Alan Ford
University of Southampton, UK

Email: ajf101@ecs.soton.ac.uk

Stig Venaas
University of Southampton, UK

Email: sv@ecs.soton.ac.uk

Intellectual Property Statement

Disclaimer of Validity

Copyright Statement

Acknowledgment

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