Project Number:	**IST-2001-32603**
Project Title:	**6NET**
CEC Deliverable Number:	**32603/UOS/DS/3.6.2/A1**
Contractual Date of Delivery to the CEC:	31st May 2005
Actual Date of Delivery to the CEC:	17th June 2005
Title of Deliverable:	D3.6.2: Cookbook for IPv6 Renumbering in ISP and Enterprise Networks.
Work package contributing to Deliverable:	WP3
Version:	1.1 (17th June 2005)
Type of Deliverable*:	R
Deliverable Security Class**:	PU
Editors:	Tim Chown (University of Southampton)
Reviewers:	Mark Thompson (University of Southampton)
Contributors:	Mark Thompson, Alan Ford, Tim Chown, Stig Venaas, Nick Lamb (University of Southampton), Christian Schild, Thorsten Küfer (University of Müntser)

* Type: P - Prototype, R - Report, D - Demonstrator, O - Other

** Security Class: PU- Public, PP – Restricted to other programme participants (including the Commission), RE – Restricted to a group defined by the consortium (including the Commission), CO – Confidential, only for members of the consortium (including the Commission)

Abstract: In this text we present the results of a set of experiments, focused on the context of enterprise networks, that are designed to be a step towards analysing how effective network renumbering procedures may be in the context of IPv6. An IPv6 site will need to get provider assigned (PA) address space from its upstream ISP. Because provider independent (PI) address space is not available for IPv6, a site wishing to change provider will need to renumber from its old network prefix to the new one. In a previous deliverable, D3.6.1, we looked at the scenarios, issues and enablers for such renumbering. Here we present results and conclusions and updated recommendations in the context of additional SOHO tests and for new experiments undertaken in enterprise (campus) renumbering. Reporting on ISP renumbering is limited; the reader is instead referred to D.3.6.1 for backbone network renumbering results.

Keywords:

IPv6 enterprise renumbering, provider assigned address space, provider independent address space

Table of Contents

Table of Figures

1. Introduction

In this text we present the results of a set of experiments that are designed to be a first step in the process of analysing how effective network renumbering procedures may be in the context of IPv6. This report is a 'successor' to D3.6.1 [6NET-D361], which focused on SOHO and backbone network aspects of the problem space.

This work was carried out jointly with funding through both 6NET [6NET] and Cisco.

An IPv6 site will need to get provider assigned (PA) address space from its upstream ISP. Because provider independent (PI) address space is not available for IPv6, a site wishing to change provider will need to renumber from its old network prefix to the new one. This is a particularly important issue for IPv6 deployment, if adopters wish to minimise their dependency on their upstream ISP. The lack of PI address space also impacts IPv6 multihoming, for similar reasons.

In D3.6.1 we reported on a number of theoretical and experimental underpinnings for the work subsequently carried out and reported here, i.e.:

- Prior art and available tools for network renumbering
- Scenarios and considerations for IPv6 renumbering
- Issues in running two prefixes on a link
- Experiments in renumbering SOHO networks
- Experiments in renumbering backbone networks
- Renumbering impact on network management and monitoring tools

We do not repeat the theoretical work in this text, nor have there been any new experiments with management and monotoring platforms to report (but the findings from D3.6.1 remain valid).

Our main focus in this document is to report the findings of experiments in enterprise renumbering undertaken at Southampton and Münster. For Münster, renumbering of DNS and server subnets/links (and applications) are considered. For Southampton, a complete enterprise renumbeing is effected, which includes tunnel broker customers of the enterprise as a form of ISP (provider) analysis. We discuss the best timing for tunnel broker customers to renumber as part of the provider renumbering in Section 3.

This work has helped contribute to and create Informational (BCP) documentation in the IETF, within the IPv6 Operations WG [V6OPS], in particular feedback for "Procedures for Renumbering an IPv6 Network without a Flag Day" [BAKER] and "Things to think about when Renumbering an IPv6 network" [THINK].

In Section 2 we describe the original set of planned enterprise experiments for IPv6 network renumbering, some of which have been achieved, others of which will be undertaken during July and August in collaboration with Cisco (as described in the Future Work section).

The main sections, Sections 3 and 4, describe the enterprise experiments at the two sites, as they report on their findings when following the renumbering procedure as defined by [BAKER] (which, to simplify, sees the link transition three stable states of original prefix in use, both prefixes in use, and finally only the new prefix in use). Both sites offer tunnel broker services, so the impact on customers of a renumbering network is also examined, as a specific SOHO case.

We then present a set of recommendations for network administrators and designers (including ISPs), vendors and application developers in Section 5.

Section 6 describes the future experiments that will be run in collaboration with Cisco.

We had originally planned to cover the full ISP renumbering scenario in this work. However, the commercial ADSL provider that we were in discussion with in late 2004 has delayed their plans for native IPv6 rollout, leaving us without a practical commercial experimentation point. The deployment done in the Greek School Network (including IPv6 over ADSL) could form the basis for future experiments, but was not ready in time for tests for this report (and we would also need to discuss and plan which sites/schools could be used in such a renumbering experiment, which would take time). Thus our 'ISP' experiments in this report are generated as the results of renumbering tunnel broker clients for an enterprise network.

We feel that an ISP can also draw good insight by looking at our SOHO and backbone results of D3.6.1 and the tunnel broker provider renumbering results presented in this report. We have also made recommendations in Section 5 that include those targetted at ISPs.

2. Potential Enterprise Renumbering Experiments

In this section we describe a potential set of enterprise experiments. In Section 6 we discuss future plans for experiments that have yet to be realised, which we hope to conduct in July and August 2005. Southampton is deploying a new (Cisco based) IPv6 infrastructure in this period, and will include a renumbering event in with the (likely) topology changes that accompany the new rollout.

Here we present experiments that other sites or researchers may wish to consider conducting.

2.1. (A) Enterprise renumbering by Baker procedure by subnet

The aim is a staged Flag Day-less approach to renumbering where isolated subnets are migrated (e.g. satellite sites, or in our case, research groups/schools) to the new prefix live and the old prefix removed perhaps before other parts of the same enterprise do so.

The experiment would check how [BAKER] scales to larger networks where sysadmin loading is such that a global renumbering en masse – however phased - is not ideal to ensure smooth transition; Is the overhead of introducing local static routes, policy, etc. for the new prefix to effectively bypass parts of the hierarchy beneficial or simply an extra burden?

This experiment is described in Section 3 for Southampton and Section 4 for Muenster below.

2.2. (B) DHCP-PD provision and subsequent controlled renumbering

Hierarchical deployment of DHCP Prefix Delegation from the enterprise edge router to the directly

connected subnets and then on to user-populated subnets lower down in the hierarchy

This would see how one could manage and trigger 'multi-staged' DHCP-PD; Further appropriateness and use of Cisco IOS Labelled Prefixes for ease of renumbering; Applicability of Baker et al procedure in this context. How is multi-layer DHCP-PD signalled? What is the impact for the hosts?

This experiment has not been carried out yet – capable hardware (and software) has been identified but is still awaited.

2.3. (C) Use of ULAs as an Enterprise Renumbering Pacifier

The idea is to try the use of Unique Local Addresses [ULA] within a site as a pacifier to the issues of live services renumbering their global prefixes. Current suggestion revolves around the use of a "non-global" DNS name suffix for local services à la multicast-dns and the efforts of the IETF zeroconf WG. Issues in discussion include participation of remote workers (e.g. MobileIPv6 or VPN users).

The test is to investigate two prefixes on a link where one is ULA and effectively site-local, plus investigate whether ULAs really are a bad idea as seems generally believed.

There are interesting issues, particularly regarding (mis-)use of DNS, RFC3484 policy and its distribution, and conformance of 3484 behaviour when ULAs are used alongside global aggregateable unicast addresses.

A new edge filtering policy is required, as is the DNS naming model consideration and local delegation within the site.

The test would see whether the impact on survivability is worth the effort, particularly given the abuse of DNS and the reluctance of the wider community to promote split-view behaviour (the alternative to a dedicated enterprise-local name suffix); also the application and stack behaviour as regards (src,dst) address selection

2.4. (D) Use of MobileIPv6 as a Flag-day enabler

There is the potential to use Mobility to mitigate the impact of renumbering. This area may be worth more investigation in due course, but MIPv6 is still in relative infancy.

2.5. Summary

The main test is the following of the Baker procedure, as reported in Sections 3 and 4 below.

Southampton plans to run experiments (B) and (C) during July and August 2005, at the same time as a new Cisco-based unified dual-stack enterprise network solution is deployed.

Experiment (D) would be interesting to explore in the longer term, when MIPv6 is more mature in deployment.

3. Enterprise Renumbering Results (Southampton)

Here we detail an IPv6 renumbering exercise undertaken according to the proposed procedure due to Baker *et al. [BAKER]* for renumbering without a flag day. It documents the stable starting state, the desired end-state, the set of obstacles and challenges impeding attaining the desired end state, and the procedure taken in achieving that state.

3.1. Introduction

The School of Electronic and Computer Science (ECS) has been delegated a /52 prefix from their parent University's allocation (2001:630:d0::/48) as awarded by the national research network, JANET. Being the only IPv6 users on campus, and without a concrete production-oriented address policy specified from the start, the School has been somewhat lax in its use of the prefix and has deployed experimental services from outside of their allocation that have since become 'production' and therefore require renumbering to be under the correct prefix.

This section documents one of a series of experiments that explore various aspects of remedying these issues, exploring different aspects of IPv6 Network Renumbering.

For the purposes of this section, the following terminology is used:

ECS School of Electronics and Computer Science IPv6 Network

UoS University of Southampton

JANET The UK National Research Network, managed by UKERNA

SLAAC Stateless Address Auto-Configuration, as per RFC2462 [RFC2462]

DHC Dynamic Host Configuration for IPv6 [RFC3315]

3.2. Landscape

ECS currently features two static uplinks to service providers, one gigabit Ethernet via UoS to JANET and thus global connectivity and a low-bandwidth commercial link (64kbps) to UUNET, which was the first native IPv6 link in the UK, but currently dormant.

Within ECS, there is a parallel IPv4/IPv6 topology enabled through the use of routed IEEE802.1q VLANs, necessitated due to the lack of IPv6 support in the core network equipment deployed [VLAN-ID]. The network's core topology is shown in Figure 3-1.

Note that the figure does not show the low bandwidth link to UUNET. That is due to the link not being used as part of the production network at the start of these exercises.

The current design philosophy of the ECS network is that each individual organisational unit within the School (e.g. research group, teaching body, business unit) have their own layer 2 VLAN, and consequently their own IP subnet. For typical user and service subnets, there is no hierarchy: they are each simple 64-bit subnets.

The School also operates a number of tunnel services, provisioning connectivity to remote sites, some of which contain hierarchy and therefore have a requirement for larger address allocations. These will form part of our 'ISP' (provider) tests for the sake of this exercise.

3.3. Desired stable state

The desired state for this experiment is to have a production and research network that is wholly within its allocated prefix from the direct upstream provider, with subnet prefixes allocated such that any future political restructuring of the ECS enterprise has minimum impact.

During these particular renumbering exercises, no key infrastructure elements will change, that is: no change of provisioning technology, no new stateful configuration elements, and no change in topology; this experiment is solely concerned with renumbering the existing network from one prefix to another.

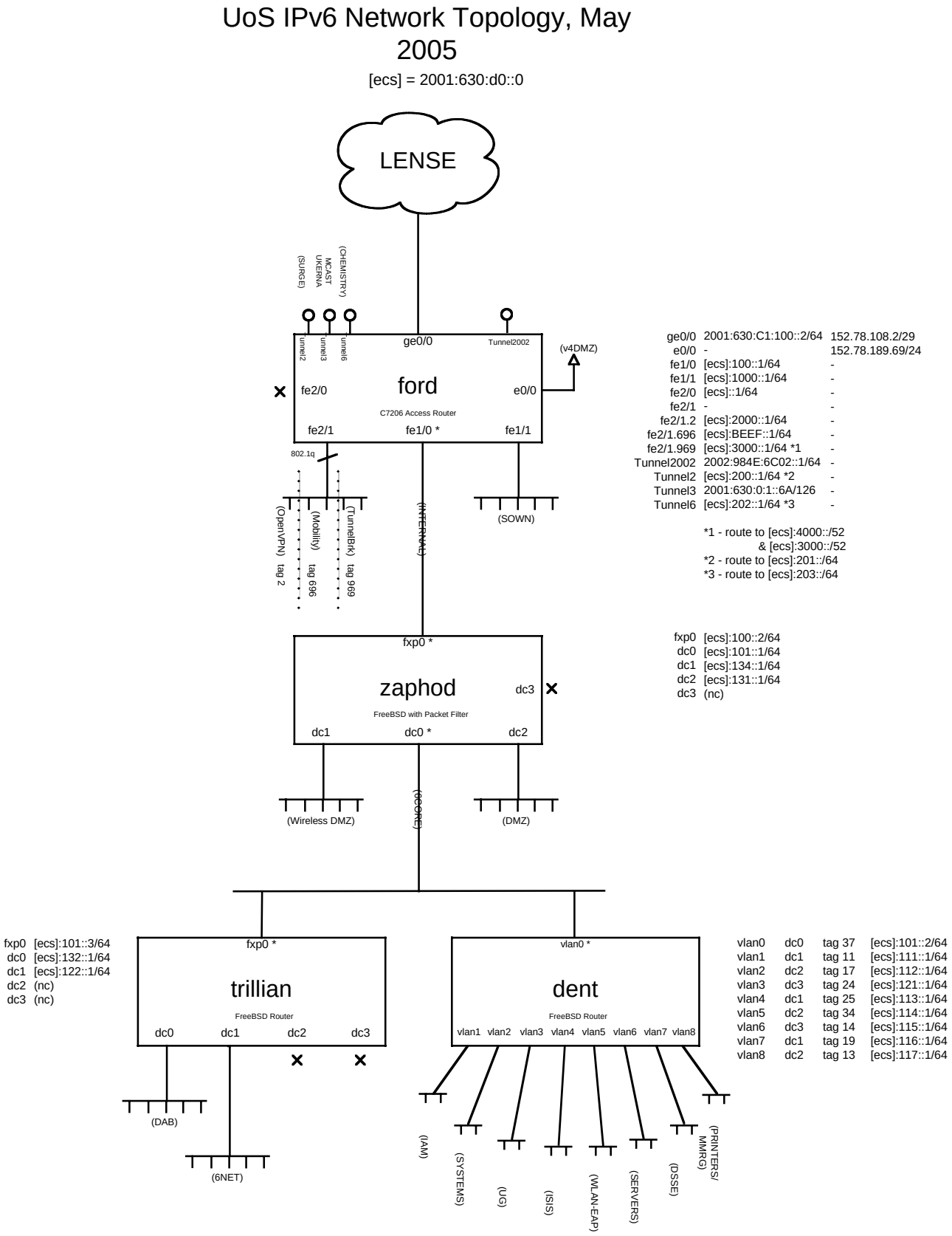

Figure 3-1: Southampton topology pre-renumbering

3.4. Obstacles

This section details a number of technical and political obstacles that need to be accommodated in reseating ECS.

3.4.1. Technical obstacles

Where the routing core of the principal ECS network is not IPv6-aware, the School has provisioned a VLAN-based parallel topology, treating the existing core as a mechanism for stringing together Ethernets and injecting IPv6 routing externally to core function. The migration to a new unified IPv4-IPv6 core is the focus of a later experiment beyond the scope of this report.

The IPv6 routing topology is shown in Figure 3-1. There is no DHC provision at present, and RIPng is used to distribute routes between three quad-NIC FreeBSD routers and the Cisco router. The packet filter on *zaphod* serves as a demarcation point between the upstream, two distinct DMZ (untrusted) zones, edge-oriented test networks and the internal 'production' network on which typical users and services reside.

The existing network topology features servers on the same subnets as client workstations. This has the implication on phased transitions to the new prefix where some subnets are at a different state of transition than others – e.g. due to scheduling obstacles as described below. There will be an ordering implication across the enterprise: subnets that house directory, naming and policy (firewalling, AAA) services will need to be at as least as late a stage as those subnets on which clients or other services reside that make use of them.

For the purposes of exploring issues in enterprise renumbering, this experiment will deliberately *not* migrate all subnets in lock-step through the Baker *et al.* procedure [BAKER]. Rather, one user-populated subnet in particular will be held back a stage at the point at which new prefix is introduced to gain insight into operational issues that would be experienced by widely distributed enterprises. Also, a satellite site provisioned by IPv6-in-IPv4 tunnels will be renumbered out-of-step with the rest of the enterprise, at that point where both prefixes are both valid and stable, as a test of 'ISP' (provider) renumbering.

3.4.2. Political obstacles

Being a technical school within the University, the demands on the network are varied and persistent. Where many enterprises may be able to have extended periods of outage outside of working hours, there are local political pressures to maintain a serviceable network out-of-hours.

This has the impact that state transitions within the renumbering procedure need to have a minimum of discernable impact on live users, and the entire episode needs to be scheduled such that it is sensitive to high-demand periods (e.g. undergraduate coursework submission events, exams, research conference events, etc.).

As hinted above, there is a risk that the School may restructure politically as research groups are realigned for various strategic reasons. The address allocation policy should reflect that risk by enabling subnet renumbering in such a way that /64s can be aggregated and still fit the allocation policy where conflation occurs, and larger aggregates supported in more extreme reorganisation events without a significant change in address allocation policy.

3.4.3. Scheduling obstacles

In larger enterprises, particularly distributed enterprises, it may be the case that an enterprise-wide renumbering episode requires a phased transition, where some satellite sites renumber at different times than others. This may be required, for example, by time-zone differences where the renumbering episode is scheduled to take place over a single working day, but that working day

being at significantly different times for different sites within the Enterprise; or perhaps where the available network engineer resource is sufficiently low that a phased migration is preferable.

As regards this particular renumbering episode, the network is deployed over a site comprising three buildings, all located on the same campus, and with sufficient administrative support that a lock-step transition can occur with all subnets synchronised within the Baker *et al.* procedure.

3.5. Strategy

The circumstances under which the renumbering of ECS is taking place means that both the old and new prefixes can be live co-incidentally and thus the "without a flag day" approach for renumbering is most appropriate. Therefore, the strategy for renumbering in this exercise is to apply the procedure proposed by Baker *et al.*.

The subsections below detail each phase of the procedure reflected against ECS live practices, with observations, impacts and particular caveats highlighted.

3.5.1. Entities of Interest

In this renumbering exercise, there are four key elements that are of particular interest: the access router, *ford* (a Cisco 7206), and three internal routers, *zaphod*, *trillian* and *dent* (each FreeBSD PC-based routers with quad-NICs, the first of these also serving as packet filter). These are shown in Figure 3-1.

3.5.2. Preparation

The preparation phase has no direct impact on the deployed network, but does require tool changes for administrative services including DNS and DHC technologies.

3.5.2.1. Obtain prefix and reverse DNS zone

The renumbering exercise will be between the existing prefix[1] and another currently unassigned from the UoS /48 prefix[2]. The School is already authoritative for the reverse zone for the University's /48 and therefore no additional delegation is required.

3.5.2.2. Work-up topology, assigning link prefixes

The target prefix is 52 bits long and therefore leaves 12 bits for sub-allocation as we intend to adopt 64-bit subnets in line with recommended best practice.

One approach that appears appropriate for address allocation within the ECS enterprise is that defined by RFC3531 [RFC3531]. Principally, the scheme enables address allocation where the extreme bits nearest the upstream and subnetting boundaries (/52 and /64 in this case) are allocated last, with a centremost expanding outwards bit pattern enacted, as detailed in Appendix A.

This fails in our case as there are two distinct categories of subnet in ECS. There are upwards of two dozen end-user /64s and a small number of subnets aggregating further hierarchy with prefixes of the order of /60 or even /56.

RFC3531 applied to the full 12-bit tier available is not appropriate for the new address space as subsequent candidate allocations are not contiguous and therefore not aggregatable. The overhead of calculating the allocation order and then selectively blocking allocations out of order for

[1] 2001:0630:00d0:0000::/52

[2] 2001:0630:00d0:f000::/52

aggregates larger than the longest prefix being allocated (in our case /64) is sufficiently high that it renders the RFC3531 approach of little benefit.

Rather, the allocation strategy taken for this exercise reflects the characterisations of the address block's intended use. That is, one set of larger aggregates for hierarchy, and one set of hierarchy-less /64s for end-user subnets.

Under `2001:0630:00d0:f000::/52`, the two left-most bits reflect the nature of subsequent allocations: 00_2 indicates end-user subnet allocations (1024 available), 01_2 indicating /60 subnet allocations (64 available), and $1x_2$ indicating larger /56 aggregations (8).

The zeroth /64 subnet is reserved for /112 point-to-point links, e.g. for tunnel end-points. The 112-bit prefix length is preferred to 127-bit on the advice of RFC3627 [RFC3627], 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. Care is needed to ensure that, should router alerts be configured on such interfaces, that the autonomous configuration bit is cleared such that interfaces do not attempt to construct 64-bit interface identifiers. The strategy taken with the /56 allocations is to leave subsequent /56s for future expantion of each sub-tier to a /55. Prior experience with tunnel provision suggests that this "allocate one, skip one" approach to /56 allocation is prudent. The resulting allocation is a straightforward approach to enterprise numbering that is flexible in that it permits topology growth, easy to understand, and manageable for administrators.

The topology by aggregation mantra of IPv6 combined with the /64-but IID recommended practice has the impact that large enterprises may have difficulty accommodating schizophrenic networks such as ECS where there is a network provision to user subnets in addition to lower tiers within their relatively small prefix boundary space (12 bits in this case). The allocation strategy devised here for ECS reflects the particular characteristics of the target network and has sufficient scope for growth.

Table 3-1 lists the new subet allocations for the ECS network. Most of the end-user subnets retain their subnet identity under the new prefix, those that have changed have been moved to cluster different kinds of subnet together, e.g. research groups, teaching clusters, etc.

Table 3-2 shows the initial allocations in the Endpoints allocation, illustrating the use of 112-bit prefixes for tunnel endpoints or point-to-point links between routers.

Old Prefix `[oldecs]|…`	Subnet Description	New Prefix `[newecs]|…`
`100`	Endpoints‡	`000::` (sets of `/112`)
`101`	6Core	`001::/64`
`102–f`	(Reserved)	
`110`	unused	
`111`	IAM	`111::/64`
`112`	Systems	`112::/64`
`113`	ISIS	`113::/64`

114	WLAN EAP	114::/64
115	Servers	102::/64
116	DSSE	116::/64
117	Printers(/MMRG)	117::/64
121	Undergrad	110::/64
122	6NET Test	103::/64
131	DMZ	104::/64
132	DAB	106::/64
134	Wireless DMZ	105::/64
200	SURGE (tunnel)	to 000::/64
201	SURGE (subnet)	200::/64
202	Chemistry (tunnel)	to 000::/64
203	Chemistry (subnet)	201::/64
1000†	SOWN	400::/60
2000†	OpenVPN	800::/56
3000†	TunnelBroker	a00::/56
4000†	HexagoTB	c00::/56
beef†	Mobility	(deleted subnet)

Table 3-1: Work-up of ECS subnets with new subnet prefixes (12-bit subnet space)[3]

Old Pair [oldecs]|…	Link Description	New Pair [newecs]|…
100::1, 100::2	ford:fe1/0 to zaphod:fxp0	::1, ::2
2000::1, 2000::2	ford:fe2/1.2 to OpenVPNrtr:eth0	::1:1, ::1:2
3000::1, 3000::2	ford:fe2/1.969 to broker:eth0	::2:1, ::2:2
3000::1, 3000::3	ford:fe2/1.970 to broker2:eth0	::3:1, ::3:2†
200::1, 200::2	ford:Tunnel2 to surge-	::4:1, ::4:2

[3] † denotes previous address allocation 'abuse' outside our proper /52 prefix. ‡ formerly internal, now utilised for point to point links. [oldecs] is 2001:630:d0:0::/52. [newecs] is 2001:630:d0:f000::/52.

202::1, 202::2	router:tun0 ford:Tunnel6 to chem-router:tun0	::5:1, ::5:2

Table 3-2: Work-up of renumbered ECS point-to-point links[4]

3.5.2.3. Ramp-down DNS Time-to-Live

ECS policy is to hold DNS resource records at a default level of 30 minutes, which is ample for the timescale of this renumbering exercise. The propagation delay for updates between primary to secondary servers is two hours, which is also adequate.

DNS nameserver glue records are currently at one day, which is fine (particularly as ECS upstream servers do not as yet support IPv6 glue).

Site MX server resource record TTLs are set to 5 minutes due to local administrative activity, and will persist at that level throughout the renumbering exercise.

Appendix B shows current ECS BIND nameserver zone file metadata, plus initial state information regarding the NS and MX records.

3.5.2.4. Ramp-up frequency for DHCP binding refresh

DHCPv6 is not deployed in ECS (due to a lack of mature client and server implementations), and will not be introduced as part of this exercise.

3.5.2.5. Observe current Router Alert configuration parameters

In preparation for later stages of this migration, it is necessary to observe the current state of router alerts that are published to end-user subnets such that the introduction of new prefixes and subsequent deprecation and then removal of old can be fit into the renumbering schedule.

There are two router architectures deployed that have end-user subnets attached, Cisco (*ford*) and FreeBSD (*dent*, *trillian*, *zaphod*).

Analysis of a host in the Mobility subnet indicates that ford is advertising prefixes with Preferred lifetimes of 604,800 seconds (7 days) and Valid lifetimes of 2,592,000 seconds (30 days), which is consistent with IOS defaults and as expected given no configuration deviating from the defaults in the configuration of the router. Analysis of the FreeBSD `rtadvd` process on *zaphod* indicates that there is no local configuration, therefore the process runs with defaults of 7 days preferred, 30 days valid.

trillian, another FreeBSD router, is currently configured such that `rtadvd` advertises prefixes to the two demilitarised subnets with validity times of 1 hour, preferred lifetime of 30 minutes.

Likewise, *dent*, the principal user subnet router, has `rtadvd` configured with 1 hour and 30 minutes respectively for validity and preferred lifetimes.

In order to satisfy stages 2, 5 and 6 of the renumbering procedure, explicit configuration of the router advertisement lifetimes will be required for those routers that currently do not have options set.

One point to note when planning a renumbering episode is that the Valid Lifetime offered in router advertisements can only be set to zero on a node if the existing lifetime is less than 2 hours, as per RFC2462 section 5.5.3 [RFC2462]. It is therefore a recommendation of this study that routers are

[4] † broker2 moved to a separate point to point link in an 802.1q VLAN rather than a VLAN of three nodes.

configured to advertise prefixes with a two hour validity time as soon as a renumbering episode is on the horizon such that prefixes can be invalidated in a timely manner. An appropriate corresponding Preferred lifetime would be 30 minutes.

With this in mind, all routers advertising prefixes in the Enterprise were configured such that their validity and preferred lifetimes in preparation for the renumbering exercise were 3,600 and 1,800 seconds respectively.

3.5.2.6. Find all the places...

As part of the preparation process, it is important to know all of the places — within administrative control and fair reason — that address literals exist as regards the network undergoing renumbering.

Focusing on core network services, the first set of services that are of importance are:

Policy

zaphod and *dent* have policy in place for packet filtering; the School principal web server has policy in place for IP-based access controls

DNS

Deployed name servers are configured to accept packets on INADDR_ANY, i.e. not specifically tied to the old prefix; resource records are discussed below

Web

Deployed web servers accept connections on INADDR_ANY (With Apache, this concerns the `Listen 80` and `NameVirtualHost *:80` configuration directives)

MTA

Deployed mail transfer agents are configured to accept connections on INADDR_ANY; MX records in DNS need to be migrated

Mail homes

Deployed mail homes do not offer IPv6 access for IMAPv4 or POP3

Compute servers

Deployed remote access servers are configured to accept connections on INADDR_ANY

Directory servers

Deployed directory servers do not offer IPv6 access for LDAPv3

Database servers

Deployed database servers do not offer IPv6 access for SQL

Collaboration servers

Deployed IRC servers are configured to accept connections on INADDR_ANY, however inter-server links are explicit address literals that will need to be migrated in later stages of the renumbering exercise. This impacts remote administrators; Deployed Jabber servers are configured to accept connections on INADDR_ANY

Multicast routers

Each of the core routers performs multicast routing for their connected subnets, with *ford* acting as rendezvous point for the enterprise. Each of the multicast routers explicitly mention *ford*'s old IP address by name, and so their configuration will need to be changed (and router restarted) once the new address is stable.

Resolver configuration

`resolv.conf` and associated files that explicitly name enterprise nameservers under the old prefix need to be updated to point at the new addresses for the servers. During the stages where both prefixes are valid, the resolvers may of course be configured to connect to either old or new addresses

3.5.2.7. Stage Summary

At this point, the revised address plan is done reflected against the desired topology and the network elements identified upon which new configuration is to be loaded.

3.5.3. Prepare network elements

The second stage of the renumbering process makes ready the routing infrastructure of the enterprise such that the new prefix is routable, but does not configure any nodes with new addresses in live service.

3.5.3.1. Policy application

Before the new prefix is made routable, it is important to ensure that security policy is updated such that no network elements are vulnerable to attack during the migration to the new prefix.

As a starting point, a block at the border router (*ford*) on all incoming traffic to the new prefix is applied ('default deny' taken to an extreme).

With the border block in place, the next step is to work through all instances of address-based policy in the School and ensure that rules are replicated for the new prefix.

One key caveat here is in policy 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". There are four combinations of endpoint that need to be considered, thus: `old` to `old`, `old` to `new`, `new` to `old`, and `new` to `new`. An example configuration snippet from FreeBSD's `pf` is included in Appendix D.

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.

3.5.3.2. Name service provision

In preparing DNS for the update, ECS already has authority for the new reverse tree in DNS as they are currently the only IPv6-capable users on campus and operate the peering onto the regional network, LENSE.

As discussed above, the time-to-live configuration data for resource records in both the forward zone for labels and reverse zone for the old prefix are such that updates will be visible by remote sites in a timely manner.

ECS, like many other enterprises, makes use of a bespoke tool for managing name service data. `hosts2zone`, a locally developed tool, combines name service data for local NIS services in addition to populating BIND zone file data for DNS.

Where such tools facilitate day-to-day administration of name services, they can introduce problems when attempting to renumber networks. For instance, the ECS systems administration team made a policy decision with their IPv4 network that no label shall be permitted to have multiple IPv4 A records bound to it, and so developed their administration tool with that policy in mind.

When IPv6 was deployed, the tool was extended to manage both forward and reverse zone data for IPv6 with the same policy implicitly imposed, the upshot being that no label can have multiple AAAA records associated with it.

Where round-robin label-based load balancing is deployed, there are overrides to the administration tool to facilitate this: place the label(s) and their associated resource records in the template and the tool will then not auto-generate data for that label when run. This "special case" provision typical to many administration tools may work for a small number of instances, but when every single AAAA record becomes a special case, the situation can become complicated and hazardous.

The approach taken for the ECS renumbering exercise has been to utilise the bespoke script to generate sample zone files (forward and reverse) for both new and old prefixes individually, then manually construct temporary zone files for deployment during those stages where nodes are multi-addressed with global prefixes[5].

3.5.3.3. Router provisioning

For each of the enterprise's routers, the new prefix is loaded such that interfaces are assigned addresses as per the allocation plan above, but any router alerts that emit from the routers do not permit nodes to generate interface addresses[6].

There being no stateful DHC deployed, concerns over offers made using the new addresses are moot in this exercise.

As regards enacting this provisioning, manual interface configuration is performed in-place safe in the knowledge that the policy application at the border will protect local network elements during this setup phase.

Configuration files (e.g. `rc.conf` and `rtadvd.conf` on the FreeBSD routers, and the `startup-conf` of the Cisco edge device) are installed with both old and new interface configuration in-place, where the old prefix is a manually configured alias address and the new prefix the primary address of the interface.

This has the impact that, should anything go wrong and the router requires a restart, the interfaces will come up in a sound state and both prefixes routable.

The set of interfaces annotated with an asterisk in Figure 3-1 participate in a RIP cloud propagating subnet routes between the routers. As the new alias interfaces were brought live, the routes could be seen as valid on the other routers.

An interesting caveat with manually bringing up the new prefix in the core is that should any erroneous interface addresses be configured on the participant routers, invalid routes will be advertised to the rest of the cloud. Deleting these invalid interfaces requires the local routing daemon to be restarted (at least in FreeBSD's case).

With the network such that all routing devices are configured to route and the various instances of policy enforcement update to include knowledge of the new prefix, the 'deny all' rule from the border filter is removed.

3.5.3.4. Stage Summary

Routing, security, and name services are now ready to accept traffic under the new prefix, but at this point no nodes are configured with addresses from the new allocation.

3.5.4. Let them bind

In this stage of the procedure, the old prefix is left alone but nodes are permitted to acquire addresses under the new prefix and so become multi-addressed.

[5] Naturally, there is now a feature request logged with the developer to cope with multi-addressing, but such implementation would not be feasible within the timescale of the renumbering episode.

[6] i.e. Router Alert Prefix Information field has the 'autonomous configuration' bit clear

No DHCPv6 is deployed, and so there is no regard for stateful node configuration.

Routers that provision subnets are now configured so that they emit router advertisements with prefix information fields that indicate that the new prefix is live; with the same prefix valid and preferred lifetimes as the old prefix, and also with the Autonomous configuration bit set in the cases of subnets on which SLAAC is to be performed.

At this point, with router advertisements being issued with a valid lifetime (i.e. greater than zero), the handful of nodes identified as being manually configured[7] with well-known IIDs have their interfaces updated to include addresses from the new prefix.

For the purposes of mimicking a satellite site's participation in the renumbering exercise, one subnet was omitted from this stage (Wireless DMZ, old subnet 134 and new subnet 105). Its new prefix will not be made live until the rest of the network has enjoyed a half-day with both address prefixes stable.

Also, a remote tunnelled satellite site that is currently having a politically invalid /52 routed toward them will be signalled of the network renumbering towards the end of the stage where both prefixes are stable. They will adopt a complete Baker *et al.* renumbering episode at that point, as detailed later.

3.5.4.1. DNS

With nodes now taking addresses under both prefixes, the enterprise DNS is updated so that resource records are served with both sets of addresses, and that the reverse DNS tree is populated with PTR records for both prefixes.

3.5.4.2. Stage Summary

At this point, both prefixes are live and nodes within the enterprise are multi-addressed. Operational checks confirm that those services bound to the unspecified address INADDR_ANY are correctly responding to connections to their new address when served by DNS or explicitly named by IP are behaving correctly.

DNS is populated with multiple addresses for each label in the forward zone, and the reverse zones for both prefixes are populated.

3.5.5. Both prefixes stable

At this time, both prefixes are stable, which gives opportunity to comment on multi-addressing in an enterprise-wide environment.

3.5.5.1. Policy success

With the caveats of Appendix D caught, policies that affect traffic to nodes under the old prefix appear to be correctly applied to nodes under their new addresses.

Correspondingly, combinations of inter-subnet communication attempts are correctly caught where a mix of old and new source and destination addresses are selected.

3.5.5.2. DNS behaviour in applications

Further to the reporting above, resolvers across all platforms correctly interact with servers within the enterprise under both prefixes when explicitly queried e.g. using tools such as `dig` or

[7] Linux, used locally for service provision, does not have avaialble a tokenised interface identifier generator as with Solaris, and therefore some well-known nodes are manually configured. Ideally, DHCPv6 would be deployed to configure these nodes - or linux support for tokenised IIDs implemented.

`nslookup`. However, the behaviour of the resolver as exposed to applications is 'interesting' across platforms, and discussed below.

3.5.5.3. Application and service effects

The following user experiences were reported with both prefixes live.

SSH host keys.

Those users that cache host keys of SSH targets — which is almost everyone, given the default configuration of SSH client applications — will receive a warning and a prompt informing them on connecting to their server's new address, which will then be cached against the server domain name.

`pim6sd`

A side effect of introducing new interfaces or addresses to interface aliases was observed with BSD's `pim6sd` multicast routing daemon. It needed to be restarted in order to recognise the new addresses. No configuration changes were needed, just a daemon bounce. Note that this will also be needed when the old addresses are invalidated in a later renumbering stage.

Web page timeouts

Some users of the network had IP-based access controls in their user accounts (out of the purview of most administrators), and therefore were not allowing access to on-line resources intermittently, depending on the address to which their browser software chose to bind. Such an intermittent problem in an application can be difficult to track down (a case for monitoring tools flagging deprecated address use).

3.5.5.4. Resolver inconsistencies

It became rapidly evident that the resolver implementations on different platforms were behaving inconsistently when multiple AAAA records were returned for a given label.

A common test across available client platforms within the enterprise has been to deploy a common version of the Firefox web browser (specifically, v1.0.4) and a test page on a multi-addressed server that displays the client and server address pair used for a given connection. For ICMPv6 address selection testing, ping and traceroute utilities were used under various conditions.

Table 3-3 summarises the results of the series of tests. 'n/a' in a column indicates that the nature of the address was fixed by the test (old or new prefix). A 'Y' in the third column for each operating system corresponds to expected behaviour, an 'N' indicating inconsistent or incorrect behaviour.

The key observation from these results is that behaviour across platforms is inconsistent, and in Linux's case particularly inconsistent with itself.

This observation is moot, though, given that the choice of address pair is not of great consequence whilst both prefixes are equally valid, particularly as policy is in place such that all combinations are covered.

3.5.5.5. Remote tunnelled satellites

With both prefixes stable and both prefixes routed over tunnels (whose IPv6 endpoints are valid both under new and old prefixes), traffic is still routed correctly to the remote satellite networks under, albeit under the old prefix.

A later section describes the renumbering exercise for one example remote satellite site enacted with both prefixes live in the main enterprise network. This particular renumbering exercise was enacted after six hours of both prefixes being stable.

Test		Linux 2.6.8			Windows XP SP2			Mac OS/X 10.4.2			Solaris 10		
		Src[8]	Dest	?	Src	Dest	?	Src	Dest	?	Src	Dest	?
Ping	Ping label	New	Either	N	Old	Old	Y	New	New	Y	New	New	Y
	Fixed old source	n/a	Either	N	n/a	New	N	n/a	Old	Y	n/a	New	N
	Fixed new source	n/a	Either	N	n/a	New	Y	n/a	Old	N	n/a	New	Y
	Target old IP	New	n/a	N	Old	n/a	Y	Old	n/a	Y	Old	n/a	Y
	Target new IP	New	n/a	Y	New	n/a	Y	New	n/a	Y	New	n/a	Y
Firefox[9]	Label	New	New	Y	New	New	Y	Old	Old	Y			
	Target old IP	New	n/a	N	Old	n/a	Y	Old	n/a	Y			
	Target new IP	New	n/a	Y	New	n/a	Y	New	n/a	Y			

Table 3-3: Address tests, both prefixes equal state

3.5.5.6. Session-based watchpoints

The following scenarios were tested with both prefixes live to observe any unanticipated side-effects of introducing the new prefix. Unexpected behaviours are flagged in-line.

Local long-lived connections

Existing connections between two nodes within the enterprise were entirely unaffected by the introduction of new DNS data for endpoints and new (additional) addresses for endpoints. New connections made after both prefixes were of equal status (both preferred and valid lifetimes) behaved as per Table 3-3.

Local short-lived connections

Any short-lived connection that was live as new prefix was advertised survived and concluded naturally. New connections made after both prefixes were of equal status behaved as per Table 3-3.

Multicast group memberships

Existing memberships continued unabated as the new prefix became available. However, `pim6sd` multicast routers will need to be restarted as the new addresses are configured on the routers.

Remote outbound long-lived connections

Any long-lived outbound connection from the enterprise to a remote (i.e. off-site) node remain unaffected. New outbound connections select source addresses in line with Table 3-3.

Remote outbound short-lived connections

Ditto.

Remote inbound long-lived connections

Any long-lived inbound connections from a remote site are unaffected by the addition of the new prefix. New connections after the addition of the new prefix and the population of DNS forward-zone appear to depend on the remote resolver (as to which address it elects to resolve the label to). Connections to either local new address are equally as valid in this stage.

[8] In our experiments Linux always chose the New address for the sources, but other iterations have shown that it may choose the old address as a source. However, whichever it chooses first, it sticks to consistently.

[9] Each of the browser tests were repeated by forcible reloading of the page in the browser to confirm the address selection was consistent

Remote inbound short-lived connections

Ditto.

3.5.6. Transition from old to new

The first step when transitioning away from the old prefix, having established that the new prefix is live and routing properly, is to reconfigure the routers to advertise the old prefix with a preferred lifetime of zero seconds, i.e. deprecate the prefix. The autonomous configuration bit and the validity time are left untouched, for nodes should still have interface ids from that prefix, and the prefix is still routable.

At the planning stage of this process, all routers with SLAAC nodes attached were configured such that their prefix preferred lifetime was 30 minutes. As a result, the worst case of a 30 minute wait is needed to ensure that all SLAAC nodes have their old address deprecated.

Test		Linux 2.6.8			Windows XP SP2			Mac OS/X 10.4.2			Solaris 10		
		Src	Dest	?	Src	Dest	?	Src	Dest	?	Src	Dest	?
Ping	Label[10]	New	Either	N	New	Old	Y	New	New	Y	New	New	Y
	Target old IP	New	n/a	N	New	n/a	Y	New	n/a	Y	New	n/a	Y
	Target new IP	New	n/a	Y	New	n/a	Y	New	n/a	Y	New	n/a	Y
Firefox	Label	New	New	Y	New	New	Y	New	New	Y			
	Target old IP	New	n/a	N	New	n/a	Y	New	n/a	Y			
	Target new IP	New	n/a	Y	New	n/a	Y	New	n/a	Y			

Table 3-4: Address tests, old prefix deprecated but addresses still in DNS

3.5.6.1. Address selection tests

With the old addresses deprecated, a series of tests were undertaken to investigate address selection on client nodes. Table 3-4 summarises the results of these tests.

Essentially, each system appears to prefer the new (preferred) prefix irrespective of length of prefix match. This is appropriate behaviour according to RFC3484 [RFC3484].

3.5.6.2. Removing old prefix from DNS RRs

Once assured that all SLAAC nodes are deprecated, and having manually marked those interfaces that are manually configured as explicitly deprecated, addresses from the old prefix can be removed from DNS resource records.

Using the latency calculation from Baker *et al.*, a time can be calculated after which it is safe to assume that no node should attempt to connect to the old address for nodes within the enterprise (save those that are using IP addresses explicitly — these should be corrected!).

In the case of the ECS enterprise, the network was left overnight with old addresses deprecated and no longer in DNS.

3.5.6.3. Service bouncing

Some services were identified in previous stages as being dependent on the old address prefix. For example, the enterprise IRC server. At this point in the procedure it is prudent to update those configurations and, where necessary, restart the services so that they come up using exclusively the new prefix.

[10] Label resolution dependent on operating system implementation, but New address consistently chosen as preferred irrepsective of preix length match

Also, remote sites that reference local IPs explicitly (e.g. remote IRC servers in the case of ECS) should now also update their configurations and restart. There should be *no* off-site reference to the old prefix in live service.

Any services that are configured by label rather than address literal may also need to be restarted. Thankfully, a survey of the ECS enterprise reveals that all services of note are configured as INADDR_ANY, and therefore do not require restarting.

At this stage, the multicast routers for each subnet are updated such that they reflect the new address of the rendezvous point under the new prefix, and restarted. The upshot here is that all existing group memberships are lost and thus it is advised that this is done out-of-hours or during a period of at-risk or other low use.

3.5.6.4. Invalidate old prefix

With all mention of the old prefix well and truly expunged from DNS resource records, the validity lifetime of the router advertisements is set to zero, invalidating the prefix on all SLAAC nodes.

At this point, all manually configured nodes also have the address manually removed from their interface configuration.

3.5.6.5. Observations

As router advertisements are received invalidating prefixes, hosts should remove the addresses under that prefix. Two hours after the validity time was set to zero — the previously last-known-good validity time expressed in a router advertisement — the addresses started to disappear from SLAAC nodes in some clients, but not others. Table 3-5 details a summary of the observations with various nodes.

Operating System	Behaviour observed with Valid/Preferred of 0/0	Compliant?
Linux 2.4.20	Immediately dropped the prefix upon receipt of 0 for validity time	N
Linux 2.4.21+[11]	Kept the prefix valid, but counted down to 0 linearly, at which point it was automatically deleted	N
Windows XP SP2	As Linux 2.4.21+	N
Mac OS/X 10.4.1	Kept the prefix valid, every time a 0 validity advertisement received, the last non-zero time used	Y
Solaris 8	As Mac OS/X 10.4.1	?[12]
Solaris 9	As Mac OS/X 10.4.1	?
Solaris 10	As Linux 2.4.21+	N
FreeBSD 4.9	As Mac OS/X 10.4.1	Y

Table 3-5: Interface tests, old prefix invalidated

The behaviour observed shows that some operating systems (correctly, in our interpretation of RFCs 2461 and 2462) reset the validity time to two hours upon receipt of a *non-authenticated* router advertisement with a validity time of less than two hours (i.e. zero) where the previous validity time was less than two hours but non-zero. Refer to clauses (e) and (f) in RFC2462 section 5.5.3 [RFC2462].

[11] Linux 2.4.21 and newer Linuxes, including 2.6.8, behaved identically.

[12] Unable to determine current lifetimes on Solaris

Those operating systems that, upon receipt of a zero validity time, continue to count down from the previous time without resetting or immediately drop the prefix, are behaving acceptably in our case, but may suffer denial of service attacks in networks where rogue nodes emitting router advertisements maliciously attempt to invalidate the local prefix.

The work-around in the case where no IPsec infrastructure is available to authenticate router advertisements is to first deprecate the old prefixes as normal, and then when it is desired that the old prefix is invalidated, simply remove the advertisement directive from the router. All nodes will then 'count down' from their existing validity time to zero due to there being no updates in router advertisements to suggest otherwise. After the last-known validity time has passed, the prefix is invalidated.

Caveats regarding timing from the table above observed, all TCP sessions that were live under the old prefix ceased ungracefully, as expected:

Local long-lived connections

Existing connections between two nodes within the enterprise dropped in all instances where one or both participants were using an address from the now invalid prefix. No new connections could be made on the old prefix, as the interface was now not configured for those addresses.

Local short-lived connections

Any short-lived connection that was live as the old prefix became invalid were dropped where one or both participants were using the old address.

Multicast group memberships

Existing memberships are based on link-local addresses to local multicast routers. However, `pim6sd` will need to be restarted once the router's interfaces have the old addresses removed.

Remote outbound long-lived connections

Any long-lived outbound connection from the enterprise to a remote (i.e. off-site) node where the source address was under the old prefix dropped. Connections anchored on the new prefix were unaffected. New connections have only one global valid address to choose from.

Remote outbound short-lived connections

Ditto.

Remote inbound long-lived connections

Any long-lived inbound connections from a remote site to an address under the old prefix dropped. Inbound connections to the new prefix were unaffected. New connections based on DNS label (that were tested) resolved to the address under new prefix. Any DNS caches "out there" that do not obey time to live data for resource records or zones may have cached the now invalid address and so may have difficulty connecting to the new addresses. However, this is outside of the enterprise's control, and common for all DNS updates irrespective of renumbering.

Remote inbound short-lived connections

Ditto.

3.5.6.6. Monitoring tools

With the old prefix deprecated and removed from DNS but the old prefix still valid and routable, it may be prudent for administrators to install traffic flow monitoring tools that can observe remote traffic coming into the enterprise on destined toward the old prefix.

Performing this on ECS it quickly became evident that, despite numerous emails to local staff, there were several external entities that maintain references to address literals within enterprise in their

DNS (i.e. outside of ECS' administrative control). This has resulted in the subnet to which those references are made (DMZ) being multi-addressed for an extra couple of days whilst those remote administrators fix their configurations. Of course, had they used CNAMES rather than AAAA resource records to reference labels that are within the administrative control of the site that is renumbering (i.e. CNAME to a label under ECS's control), then the multi-address prolonged state would not be necessary.

Certainly new SYN packets at the head of incoming TCP sessions should be flagged as a candidate for DNS cache misbehaviour or address literal abuse.

One such example tool in ECS has been to observe NetFlow traffic flows at the externally-facing interface (*ford*:GigabitEthernet0/0) and to specifically watch for traffic destined toward the old prefix.

Other points of inspection could be any Intrusion Detection Systems, such as Snort, that may be deployed, or in the case of ECS, the head of the interior routing, *zaphod* (although in the deployed topology, that would not catch traffic destined toward the tunnelled services off of *ford*).

3.5.6.7. Stage summary

At this point, no instances of the old addresses exist on end nodes, in the DNS, and hopefully in local configuration data. All services that are live are either address-agnostic, or listening explicitly to the new addresses.

The only place where the old prefix is still valid is in the routing core of the enterprise.

3.5.7. Remove old prefix

When the old prefix is removed from DNS and invalidated for all nodes, the alias addresses on the router interfaces can be deleted by hand and the configuration files updated to delete references to the old prefix.

Were ECS not authoritative for the reverse delegation in DNS and the vacated address block, it would at this point surrender them to the appropriate registries.

In anticipation of the renumbering process about to complete, DNS TTLs are ramped back up to their operationally stable level. If DHCPv6 were deployed, then the binding refresh frequency would also be ramped down at this point.

3.5.7.1. Stage summary

After this stage is completed, the old prefix is no more and the enterprise wholly configured and operational under the stable new prefix.

Where it is possible that the upstream provider still routes the old prefix toward the enterprise (as is the case here), it is possible to keep the traffic or flow monitors live such that it is possible to determine what remote entities still possess invalid address data for the enterprise, whether through misbehaving DNS caches, or explicit address literal misuse.

3.6. Renumbering a tunnelled satellite site

This section details a separate renumbering experiment that was staged with the enterprise-proper held at the point where both prefixes were stable, in DNS, and routed.

It represents some of the issues faced by an ISP renumbering its own customers, and includes recommendation as to when to effect the customer renumbering (without a flag day).

A small SOHO network that obtained its IPv6 connectivity via a 6in4 tunnel from ECS acts as a useful case study for renumbering a satellite site. The procedure followed was similar to that in the

main enterprise site, but enacted in a much smaller time, for convenience, and also because the site does not contain any critical systems.

The tunnel was configured on the tunnel server to route both the old prefix (2001:630:d0:3300::/60) and the new prefix (2001:630:d0:fa00::/60) to the IPv4 endpoint. The tunnel endpoint was a Linux machine: the firewall rules were duplicated for the new prefix, and new addresses were added to the tunnel endpoint and internal interfaces.

At this stage, the router advertisement daemon (radvd) was only giving out the old prefix, with a validity of 604800 (one week), and preferred lifetime of 86400 (one day).

Since the DNS for the end site was outside its control, new addresses were added to the DNS in good time. Once this was configured, radvd was reconfigured with both prefixes, with validity of 3600 (one hour) and preferred lifetime of 1800 (half an hour). Due to the requirements in RFC2462, the validity dropped to two hours on all active network machines, because of the previous long lifetimes. After an hour, all had stabilised to the 3600/1800 values.

There seemed to be some evidence on FreeBSD that when the old prefix had a longer lifetime, it was always chosen as the source address, if there was otherwise nothing to differentiate it with the alternatives. When both prefixes had the same lifetime, one appeared to be arbitrarily chosen for each connection. Similar behaviour was observed in Windows XP SP1 and Mac OS X 10.4.

Linux chose one source address and stuck to it for all connections, irrespective of the destination address chosen. Raw addresses would enact longest-matching prefix for source address selection on all other operating systems tested. When DNS lookups were used, the first one returned from DNS (seemingly arbitrary) was used, and the same source address selection was applied. This was tested through the use of ping and traceroute, as well as SSH sessions and HTTP to our test webserver.

When the prefix was deprecated (valid 3600 / preferred 0), all behaved correctly: all OSs chose to use the preferred prefix only for outgoing communication, unless responding to incoming on the old address. It obviously has no knowledge of remote address deprecation, so chooses the first returned from DNS, but will always communicate from the local preferred address.

Finally, the address was invalidated (0/0), the validity was counted down from its current value of 3600. Upon reaching zero, all connections died and the old addresses were removed from all hosts.

The local mail server is the only non-SSH server running on IPv6, it is run via OpenBSD *inetd*, and therefore handled the changes of address just fine.

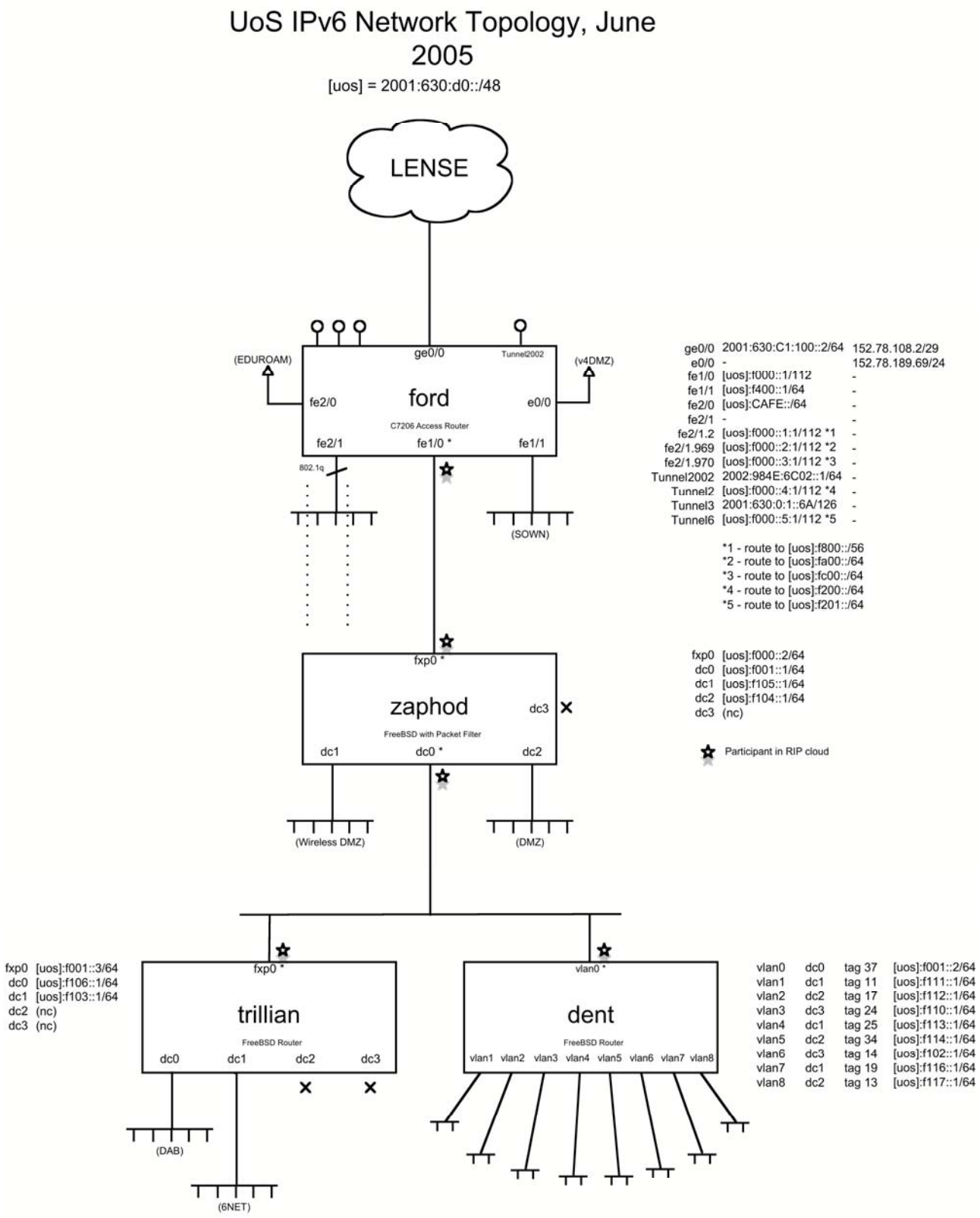

Figure 3-2: Southampton topology post renumbering

3.7. Resulting Recommendations

This activity suggests the following recommendations for good practice when undertaking enterprise-scale renumbering exercises.

3.7.1. Network Administrators and Designers

The recommendations are:

112-bit prefixes for point-to-point links

We have used /112 size prefixes for point-to-point links. We recommend that enterprises do not 'waste' /64s just for such networks

Prepare for growth where possible

We used a 'use one, skip one' approach for allocating address blocks to tunnelled satellite sites, currently allocated /56s. By allocating every other block first, there is scope to double the size of the subnet's allocation (i.e. route a /55) should the satellite site grow sufficiently, without renumbering all the other remote satellites by moving them around. We recommend that enterprises plan for growth in their address plan where there is scope to do so

Block the new prefix at the border until ready

Our filter policy is to block the new prefix at the edge of the enterprise until that time at which it is completely routable internally and all services secure.

DNS Authority

We noted a feature when renumbering the ECS enterprise where literal addresses within ECS' delegation were present in DNS resource records for domains over which ECS has no authoritative control — as is often the case in enterprises that host web servers and application servers on behalf of collaborators and customers. 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.

Use new nameservers as soon as possible

It is recommended that administrators switch over to the new addresses for nameservers in all client resolvers as soon as the new prefix is stable. This is so that no-one is caught out when invalidating the old prefix — applications such as SSH may stall or worse refuse to authenticate when the node being connected to is unable to contact configured resolvers.

3.7.2. Operating System and Router vendors

The recommendations are:

Token-based addresses

Modern Solaris versions support a feature by which the interface identifiers for a node can be tokenised (i.e. specified explicitly without relying on SLAAC). Where router advertisements heard with prefix information blocks with the autonomous configuration bit set, that token is concatenated to the advertised network prefix to make a global address. This removes the need to maintain full hard-coded literals, and therefore aids in the renumbering process. This recommendation also applies, for example, to router vendors. It is recommended that other stack developers follow this lead and implement this much appreciated feature.

Invalidation of prefixes

We observed inconsistent behaviour as regards the invalidation of SLAAC-enabled prefixes in various operating systems. Specifically, when router advertisements are received with a prefix

validity time of zero (when already within the two hour minimum window), nodes should reset the validity time to two hours unless the advertisement is authenticated. It is recommended that operating system developers correctly implement prefix invalidation.

Prefix invalidation without IPsec infrastructure

Further, where no IPsec infrastructure is available to authenticate router advertisements such that a zero validity lifetime is permitted by standards compliant nodes, the recommended practice for invalidating a prefix is to deprecate the prefix as usual and then once confident that all nodes are deprecated, remove the prefix from the advertisement set on the routers. Nodes will then count-down their validity time to zero and invalidate the prefix correctly.

3.8. Summary

The renumbering experiment for the Southampton enterprise highlighted a number of interesting issues that warrant further investigation. Overall the procedure went reasonably well. A number of the 'caveats' would have happened with IPv4 renumbering also.

Further planned experiments are described in Section 6.

4. Enterprise Renumbering Results (JOIN)

An enterprise network is a heterogeneous network consisting of some edge routers, several servers with essential services like email, web, etc. and various user desktop computers. Renumbering this kind of network involves a lot of work because all services configured with IPv6 addresses need to be manually edited and updated. The central task is to update the nameserver configuration which has to be done in several steps with regard to lifetimes of the information.

This paragraph describes the renumbering of an enterprise network at the example of the University of Münster's Intranet.

4.1. Environment

The university's currently used prefix is 2001:638:500::/48. The addressing plan is shown in Figure 4-1.

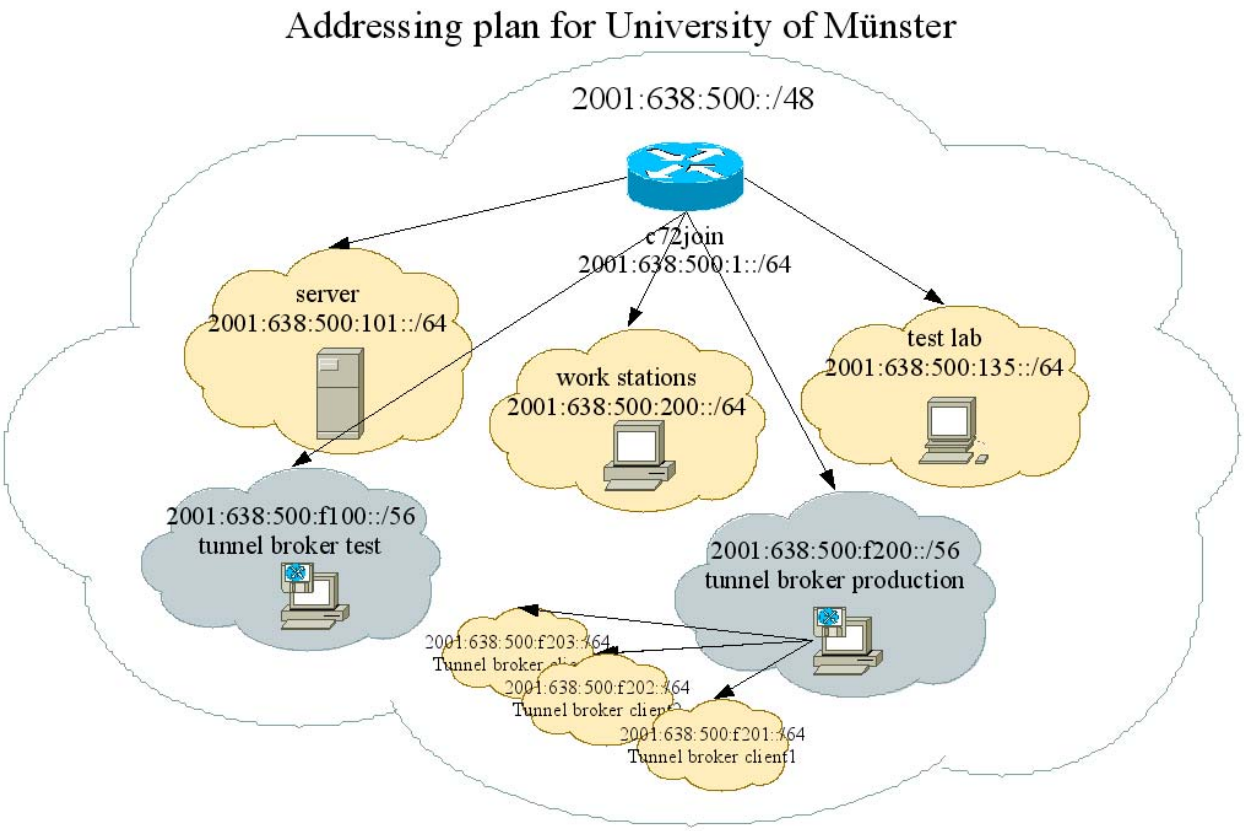

Figure 4-1: Muenster's IPv6 addressing plan

Currently the network consists of 9 routers/switches (8 of them in the test lab), 17 servers (10 in test lab), 5 workstations and 4 concurrent tunnel broker users.

4.1.1. Why renumber the network?

The university applied for a /32 prefix as an LIR, the assignment from RIPE is still to be received. Therefore we cannot renumber the whole /48 prefix but only several subnets. The most interesting subnets are servers and workstations which are going to be renumbered to the new prefixes 2001:638:500:131::/64 and 2001:638:500:132::/64.

4.2. How the network is to be renumbered

4.2.1. Preparation, usage of ULA

In theory the use of ULAs could be a useful idea during renumbering of large sites to keep up internal network connectivity for long-lived sessions (e.g. SSH, Mail, VoIP). This requires the use of DNS together with ULA in the intranet. In our case we felt that the introduction of ULAs would be too much additional work. Usage of ULAs allows for renumbering with a flag day where internal network communications (using the ULAs, which are preferred over globals for internal use) are not affected by renumbering and only outside connections have to be reestablished. This is especially useful in single-site networks (e.g. SOHO) but adds additional complexity to the network administration (because of additional DNS configuration). In an Enterprise network the introduction

of ULAs would lead to a big additional effort, including the need to manage and monitor both sets of prefixes (even where subnets are congruent) and run a two-faced DNS.

4.2.2. Updating DNS reverse zone

In this stage we are adding new reverse zone entries and reducing the time to live of the old entries.

```
; Clients #NEU#

a.c.0.e.2.8.e.f.f.f.5.7.4.0.2.0.2.3.1.0     IN  PTR lemy.ipv6.uni-muenster.de.

b.d.d.e.a.2.e.f.f.f.9.5.0.d.2.0.2.3.1.0     IN  PTR flipper.ipv6.uni-muenster.de.

a.d.5.b.0.5.e.f.f.f.8.1.0.e.2.0.2.3.1.0     IN  PTR thora.ipv6.uni-muenster.de.

9.6.6.0.7.2.e.f.f.f.4.4.2.0.2.0.2.3.1.0     IN  PTR tifflor.ipv6.uni-muenster.de.

1.d.f.c.c.4.e.f.f.f.a.5.0.1.2.0.2.3.1.0     IN  PTR kummerog.ipv6.uni-muenster.de.

4.d.d.0.3.b.e.f.f.f.5.9.a.0.2.0.2.3.1.0     IN  PTR quasimodo.ipv6.uni-muenster.de.

0.4.3.6.5.c.e.f.f.f.a.5.0.1.2.0.2.3.1.0     IN  PTR melbar.ipv6.uni-muenster.de.

5.d.1.9.3.4.e.f.f.f.a.d.0.5.2.0.2.3.1.0     IN  PTR gucky.ipv6.uni-muenster.de.


; Server #NEU#

1.2.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.3.1.0     IN  PTR mercant.ipv6.uni-muenster.de.

3.5.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1.3.1.0     IN  PTR rorvic.ipv6.uni-muenster.de.

f.5.c.b.4.0.e.f.f.f.b.4.0.e.2.0.1.3.1.0     IN  PTR tmpntpsrv.ipv6.uni-muenster.de.

f.7.a.9.9.2.e.f.f.f.1.8.0.e.2.0.1.3.1.0     IN  PTR fellmer.ipv6.uni-muenster.de.

f.b.f.f.8.2.e.f.f.f.1.8.0.e.2.0.1.3.1.0     IN  PTR ellert.ipv6.uni-muenster.de.

b.1.b.9.9.2.e.f.f.f.1.8.0.e.2.0.1.3.1.0     IN  PTR zwiebus.ipv6.uni-muenster.de.

a.f.c.7.2.c.e.f.f.f.0.2.0.0.a.0.1.3.1.0     IN  PTR ovaron.ipv6.uni-muenster.de.

6.c.7.3.4.2.e.f.f.f.1.8.0.e.2.0.1.3.1.0     IN  PTR tolot.ipv6.uni-muenster.de.

d.f.6.3.4.2.e.f.f.f.1.8.0.e.2.0.1.3.1.0     IN  PTR corello.ipv6.uni-muenster.de.
```

4.2.3. Adding new prefix

The new prefix is added and announced.

Current Cisco IOS configuration:

```
interface GigabitEthernet0/0.200
```

```
 description RZMITARB
 encapsulation dot1Q 200
 no ip proxy-arp
 no ip route-cache
 no ip mroute-cache
 no snmp trap link-status
 ipv6 address 2001:638:500:200::/64 eui-64
 ipv6 enable
 ipv6 dhcp relay destination FF05::1:3
!
interface GigabitEthernet0/0.236
 description JOINSRV
 encapsulation dot1Q 236
 ip address 128.176.191.4 255.255.255.224
 no ip proxy-arp
 no ip route-cache
 no ip mroute-cache
 no snmp trap link-status
 ipv6 address 2001:638:500:101::/64 eui-64
 ipv6 enable
!
```

We now add the new prefix for each VLAN with another `ipv6 address` command. The new entries appear in the IPv6 neighbor cache.

```
C72JOIN#show ipv6 neighbors gigabitEthernet0/0.200
IPv6 Address                              Age Link-layer Addr State Interface
FE80::2E0:18FF:FED8:6B4A                   32 00e0.18d8.6b4a  STALE Gi0/0.200
2001:638:500:132:2E0:18FF:FED6:28E7        24 00e0.18d6.28e7  STALE Gi0/0.200
FE80::211:2FFF:FEE1:D4B5                   43 0011.2fe1.d4b5  STALE Gi0/0.200
2001:638:500:200:210:5AFF:FE4C:CFD1         0 0010.5a4c.cfd1  REACH Gi0/0.200
FE80::210:60FF:FEF1:D4E9                   84 0010.60f1.d4e9  STALE Gi0/0.200
```

```
2001:638:500:132:210:5AFF:FE4C:CFD1          0 0010.5a4c.cfd1  STALE Gi0/0.200

2001:638:500:132:211:2FFF:FE76:5BEC        217 0011.2f76.5bec  STALE Gi0/0.200

2001:638:500:132:2E0:18FF:FED8:6B4A          2 00e0.18d8.6b4a  STALE Gi0/0.200

2001:638:500:200:204:75FF:FE82:E0CA          0 0004.7582.e0ca  DELAY Gi0/0.200

2001:638:500:132:211:43FF:FE0D:BB31        239 0011.430d.bb31  STALE Gi0/0.200
```

We use the `ipv6 nd prefix` command for the transition to the new prefix. In allowing a not too short time for transition new connections should prefer the IPv6 address with the longest lifetime and the amount of failing connections should be reduced.

```
C72JOIN(config-subif)#ipv6 nd prefix 2001:638:500:132::/64 infinite infinite

C72JOIN(config-subif)#ipv6 nd prefix 2001:638:500:200::/64 at 12 May 2005 16:00 12 May
2005 16:00
```

Unfortunately the Linux kernel only considers the scope and validity of an IPv6 address and not the duration of validity. This means the new IPv6 address with infinite lifetime is only selected by chance. Therefore it is necessary to set the preferred lifetime immediately to zero, e.g. with setting the end of preferred lifetime to now.

```
C72JOIN(config-subif)#ipv6 nd prefix 2001:638:500:200::/64 at 12 May 2005 16:00 11 May
2005 16:00
```

```
Router advertisement from fe80::208:e2ff:fe0e:2008 (hoplimit 255)

Received by interface eth0

        # Note: {Min,Max}RtrAdvInterval cannot be obtained with radvdump

        AdvCurHopLimit: 64

        AdvManagedFlag: off

        AdvOtherConfigFlag: off

        AdvHomeAgentFlag: off

        AdvReachableTime: 0

        AdvRetransTimer: 0

        AdvDefaultPreference: medium

        AdvSourceLLAddress: 00 08 E2 0E 20 08

        AdvLinkMTU: 1500

        Prefix 2001:638:500:101::/64
```

```
                AdvValidLifetime: 83747

                AdvPreferredLifetime: 83747

                AdvOnLink: on

                AdvAutonomous: on

                AdvRouterAddr: off

        Prefix 2001:638:500:131::/64

                AdvValidLifetime: infinity (0xffffffff)

                AdvPreferredLifetime: infinity (0xffffffff)

                AdvOnLink: on

                AdvAutonomous: on

                AdvRouterAddr: off
```

4.2.4. Updating server applications

Configuration of server applications is revised and updated to the new prefix. During JOIN's IPv6 project time it has tested and setup the following server applications:

a) Nameserver (*ns.ipv6.uni-muenster.de*)

Renumbering of the name server is highly critical because a lot of things depend on it. In particular you have to inform third parties because DNS information is transferred from and to other zones.

b) Apache web server (*www.ipv6.uni-muenster.de*)

We have to pay attention to the `Listen`, `Allow from`, `NameVirtualHost` and `VirtualHost` configuration directives. Without an IPv6 address in the `Listen` option Apache listens on all addresses, as in our case, but we still have to update the other options manually.

c) ProFTP ftp server (*ftp.ipv6.uni-muenster.de*)

The ProFTP daemon is started via *xinetd* which listens on all IPv6 addresses. For the definition of transfer groups hard-coded IPv6 addresses are used. These have to be edited during renumbering. The FTP server has a static IPv6 address which has to be changed manually. Additionally JOIN's FTP server is an official mirror for several sites (e.g. Redhat, Suse, NetBSD, Mozilla, …) which have to be informed partly because they have entries in their name servers.

d) Mail server (*mail.ipv6.uni-muenster.de*)

The IMAP daemon is started via *inetd* which listens on all IPv6 addresses. There is nothing to configure.

e) Inn news server (*news.ipv6.uni-muenster.de*)

The INN daemon can be configured to listen on specific addresses with the `bindaddress6` configuration option in *inn.conf*. If not specified as in our case INN listens on all addresses. In addition to that stunnel (a universal SSL tunnel) is used for secure NEWS. Recently IPv6 support has been added but stunnel binds to a specific address that has to be configured in stunnel.conf.

f) OpenMCU conferencing server (*openmcu.ipv6.uni-muenster.de*)

OpenMCU can be configured to listen on specific addresses with the `interface` configuration option in openmcu.ini. If this value is set to :: OpenMCU listens on all addresses. There is nothing to configure in our case.

g) Jabber server (*jabber.ipv6.uni-muenster.de*)

There is no need to update the configuration or restart the Jabber server. It binds to all IPv6 addresses and recognizes the new prefix correctly. However established TCP connections with the old IPv6 addresses stay and are not removed by the kernel together with the addresses. This is a possible bug with the Linux kernel.

h) Tunnel broker server (*zwiebus.ipv6.uni-muenster.de*)

There is nothing to do for the server software (OpenVPN) because the tunnel broker listens on IPv4. The routes on the Cisco router to the tunnel broker subnets have to be updated to the new IPv6 address of *zwiebus*. Additionally since routing is enabled on this server RAs are ignored and we have to change the (static) IPv6 address manually.

i) NTP time server (*time.ipv6.uni-muenster.de*)

This is a network time server from Meinberg running with a special Linux 2.4 and NTP daemon. There are no hard-coded IPv6 addresses and there is nothing to configure.

4.2.5. Stable use of both prefixes

The use of IPv6 addresses of both prefixes works throughout the network.

4.2.6. Updating network monitoring and security application

We have to update the hard-coded IPv6 addresses in the configuration for our server monitoring application Argus. Additionally firewall rules and other security measures like access control lists (ACLs) have to be updated.

4.2.7. Updating DNS pointers

Updating name entries with new IPv6 addresses.

```
;; Stable Neu #131

ns              IN  AAAA        2001:638:500:131::53

rorvic          IN  CNAME       ns

mercant         IN  AAAA        2001:638:500:131::21

v6serv01        IN  CNAME       mercant

ftp             IN  CNAME       mercant
```

```
rsync           IN  CNAME       mercant

ovaron          IN  AAAA        2001:638:500:131:a00:20ff:fec2:7cfa

ntp6            IN  CNAME       ovaron

ntp             IN  CNAME       ovaron

mail6           IN  CNAME       ovaron

mail            IN  CNAME       ovaron

tolot           IN  AAAA        2001:638:500:131:2e0:81ff:fe24:37c6

www             IN  CNAME       tolot

tmpntpsrv       IN  AAAA        2001:638:500:131:2e0:4bff:fe04:bc5f

time6           IN  CNAME       tmpntpsrv

time            IN  CNAME       tmpntpsrv

fellmer         IN  AAAA        2001:638:500:131:2e0:81ff:fe29:9a7f

openmcu         IN  CNAME       fellmer

ellert          IN  AAAA        2001:638:500:131:2e0:81ff:fe28:ffbf

news            IN  CNAME       ellert

corello         IN  AAAA        2001:638:500:131:2e0:81ff:fe24:36fd

zwiebus         IN  AAAA        2001:638:500:131:2e0:81ff:fe29:9b1b

;
```

4.2.8. Removal of old prefix

The old prefix is removed from the interface and not announced any more. This happens automatically due to the `ipv6 nd prefix` command. Only static IPv6 addresses have to be changed manually. The still remaining `ipv6 nd prefix` command can be removed from the router configuration as well as the `ipv6 prefix` command which however does not send RAs in the presence of the `ipv6 nd prefix` command but configures the router interface with the old IPv6 address.

4.2.9. Removing old DNS pointers

Next comes removal of old reverse zone and DNS entries. Duplicated reverse zones and old DNS entries are removed from the nameserver configuration.

4.3. Summary of results

Renumbering an enterprise network has some difficulties. At first there is the name server which has to be updated several times with new information and lifetimes which slows down the potential speed of the renumbering process. A big problem is renumbering the nameserver itself

because the resolver of all clients has to be informed about this, not to speak of partners and secondary nameservers that have to be informed too. In the first case DHCPv6 could be very useful. An alternative is to put the nameserver in a different subnet which is not affected by renumbering. A significant amount of time is taken up to handle a lot of services running in an enterprise which have to be checked and possibly reconfigured with new addresses and prefixes, in addition to any firewall and security rules that are applied.

5. Updated Recommendations to aid IPv6 Renumbering

The recommendations cited here are updated from D3.6.1 in the light of experimental results in provider (tunnel broker) and enterprise renumbering.

This section comprises a set of recommendations for ISPs, Network Designers, Administrators, Application Developers, and Operating System and Stack developers such that their day-to-day activities permit network renumbering events without significant hindrance.

These recommendations have arisen as a result of research into three different network models. At the time of writing, the Small-office/Home-office (SoHo) scenario and Enterprise network scenarios have been the primary concerns, with some consideration of ISP and Core/Exchange networks.

Some of the recommendations may be considered ‘contentious’, or perhaps ‘religious’; our aim here is to suggest measures that could ease network renumbering procedures, and provoke discussion of the ideas and issues by the communities that can have an impact.

5.1. Network Designers

5.1.1. Tokenised Interface Identifiers for Core Services

‘Tokenised’ well-known addresses offer the ability to have known addresses for core services without the overhead of running DHCPv6.

Enabled through the *token* socket I/O directive in Solaris, `SIOCSLIFTOKEN`, specifying a token on an interface before it is brought up will mean that the token is used for the interface identifier part of statelessly autoconfigured address as opposed to the EUI-64 address derived from the NIC’s MAC.

The `FE80::/10` link-local address will still be auto-generated using the EUI-64 address, however any Router Advertisement received in compliant nodes will be prepended to the token to form a node address of `<prefix>::<token>`. For example, specifying a token of 25 might be appropriate for a site’s (or subnet’s) SMTP relay.

5.1.2. 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 Enterprise /64. This simplifies policy-based filtering and is less wasteful of address space than using /64s everywhere.

The 112-bit prefix length is preferred to 127-bit on the advice of RFC3627, 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.

5.1.3. 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 centremost bitset growth as described in Section 3.3 of RFC3531, 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.

5.1.4. Network Architecture Protection

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 NAT devices obscure identity - 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 NAT removal, for example to provide true end-to-end connectivity, to make simple the gateway between site and Internet, to encourage 'considered' policy as regards secure access rather than the weak and dangerous defence of hiding behind a NAT.

The Network Architecture Protection work [NAP] by the IETF discusses how NAT's 'benefits' can be achieved and surpassed using IPv6 and global addresses.

5.1.5. NON-recommendation: Unique Local Addresses

Initial work studying protocol enablers of IPv6 as regards network renumbering suggested that unique local addresses would allay some of the issues observed by sites that have global Internet connectivity but suffer repeated renumbering events, e.g. due to different upstreams per episode (e.g. community wireless networks) or other networks that may, with IPv4, have benefited from Provider Independent address allocations.

ULAs appear to lend themselves particularly well for long-lived sessions whose nature is intra-site, for example local file-store mounts over TCP-mounted NFS: With clients using ULA source

addresses to mount file-store using the ULA of an NFS server, both client and server can have their global routing prefix renumbered without consequence to ongoing local connections.

However, ULA deployment incurs several technical and administrative difficulties, including split-view DNS (so that internal clients are assured to resolve to locally-addressed instances of services independent of current global prefix availability or applicability); source and destination address selection policy distribution and compliant RFC3484 implementations; filter policy application when ULAs leak into global DNS or across administrative boundaries; application awareness of applicability of addresses, particularly as ULAs are syntactically indistinguishable from global aggregatable addresses.

Further experimentation with ULAs as regards operational network use where global addresses are available, and as a potential renumbering enabler, are on-going concerns.

5.2. Network Administrators

5.2.1. Block the new prefix at the border until ready

A prudent filter policy in preparation to renumbering to a new prefix is to block all traffic to what will be the new prefix at the ingress point until all internal policy enforcement interfaces and routers are configured and ready for traffic (i.e. the stage where the new prefix is internally routable).

5.2.2. Policy rule replication where both prefixes valid

Following on from the previous recommendation, it is important that all instances of policy concerning address or network literals (e.g. packet filters and firewalls) have their correct rule combinations in-place to cope with multi-addressed nodes in the network.

One key caveat here is in policy 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”. There are four combinations of endpoint that need to be considered, thus: `old` to `old`, `old` to `new`, `new` to `old`, and `new` to `new`.

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.

There is dilemma here in that the recommendation concerning the use of symbolic names to identify elements in the network may not be appropriate, e.g. for specifying firewall 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.

5.2.3. In DNS, stick with authorities — use CNAMEs

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.

5.2.4. Use new nameserver addresses as soon as possible

It is recommended that administrators switch over to the new addresses for nameservers in all client resolvers as soon as the new prefix is stable. This is so that no-one is caught out when invalidating the old prefix — applications such as SSH may stall or worse refuse to authenticate when the node being connected to is unable to contact configured resolvers.

5.2.5. Avoid literals at all costs

It is strongly recommended that administrators avoid using hard-coded address literals wherever possible.

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.

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 start-up 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 in service load balancers (e.g. pool of addresses for nodes offering mirrored services)
- 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
- Network monitoring software configuration files
- Network performance monitoring software configuration files (local and remote)
- PIM-SM Rendezvous Point address

Some hard-coded IP address information will be held in remote locations, e.g. remote firewalls, DNS glue, etc., increasing 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.

Note that relying on labels for endpoint identification during possible renumbering can be problematic should administrators be lackadaisical in planning for the renumbering episode by stepping down DNS Resource Record Time To Live data.

Some router vendors, e.g. Cisco, provide a prefix labelling option ("IPv6 General Prefix" in IOS parlance, see below) that enables network administrators to refer to a prefix by name rather than repeatedly stating the literal address part throughout the router configuration, essentially minimising the number of elements that have to be modified when renumbering.

5.2.6. Ramp up/down service data well in advance of any anticipated renumbering episode

On routers, it is vital to remember to step-down the Router Advertisement Validity and Preferred times, but with care to ensure that step-down for Validity time is advertised ahead of 2 hours before the renumbering episode is to take place ([RFC2462], Section 5.5.3(e)]). This enables stacks to deprecate the old prefix in a timely manner and then subsequently remove the old prefix and associated node-maintained route information at the appropriate time (i.e. as the old prefix genuinely becomes unavailable, not afterward).

It is also recommended that, where possible, administrators plan to 'remove' the old prefix from operational use well in advance of any hard limit imposed, e.g., by up-stream providers.

Should a site be using DHCPv6 for stateful address configuration, be sure to manage the lowering of lease times (increasing Binding Refresh intervals) and prepare for Reconfigure messages [RFC3315, Section 19.1] to be sent (individually) to bound clients.

Where the site has node global address data in the DNS, the Time To Live data bound to resource records concerning nodes that are to be renumbered should be decreased and zone data cache and refresh intervals ramped such that external nodes will be assured of resolving (authoritatively) correct data post-renumbering.

5.2.7. Monitor flows on both old and new prefixes

Configuring monitoring tools to alert on traffic flows toward deprecated or even removed prefixes (where border routers are still on the routing path for the old prefix) offer an invaluable way of identifying what configuration artefacts remain out of date during or following a renumbering episode. For example, malconfigured or mis-cached DNS resource records, address literals stored in configuration files off-site or out of the administrative control of the renumbered enterprise, on-going sessions that have persisted beyond address deprecation and/or validity.

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.

5.2.8. Use of Anycast within a site; or globally

Beyond the use of tokenised interface identifiers, one recommendation that is emerging is the use of Anycast for core services, whether locally as well-known site-wide anycast addresses independent of allocated provider-assigned addressing, or globally using globally reserved anycast addresses such as used in IPv4 for the 6-to-4 transition mechanism.

For example, specifying a globally recognised anycast address for DNS servers as a shipping default (e.g. in operating systems, or as OS “kickstart” images) would offer a ’least effort guaranteed functionality’ for nodes, with address resolution being operational without additional configuration or the presence of a stateless DHCPv6 service locally, particularly when combined with stateless address auto-configuration.

Using anycast for DNS resolver address throughout the internet will provide networks with an assured-default resolver, which may be used as either a fall-back in the case of locally configured resolvers being uncontactable (e.g. due to administrators invalidating old name server addresses before propagating configuration details for the nameservers’ new address), or as the preferred address to use explicitly for a resolver — particularly useful in cases where no split-view DNS is needed and any local nameservers could participate in the anycast routing within the enterprise.

5.3. ISPs

The recommendations above apply to ISPs, and in addition there are two policy oriented recommendations that should be considered

5.3.1. Allocate static prefixes to customers

In IPv4, many SOHO users get a dynamic IPv4 global address on each connection, even with ‘always-on’ technologies like ADSL. For IPv6, customers a re likely to want to run services on the globally routable address block allocated to them. To avoid renumbering their network, customers should be allocated a static IPv6 prefix wherever possible on each connection.

5.3.2. Allocate fixed length prefixes to customers

The recommendation of RFC3177 is that a ‘site’ network is allocated a /48 prefix by its ISP. To avoid a site having to consider topological issues when renumbering or changing provider, we recommend that ISPs allocate consistent prefix lengths to customers. While there is currnetly some discussion as to the content of RFC3177, it does represent best practice now, so sites should receive /48 prefixes wherever possible.

The problem is greater where a site moves from a shorter prefix to a longer one, e.g. from a /48 to a /60. The implication for an ISP however is that if it has more than 2^16 customers, using static, fixed-length /48 prefixes, it will need more than the ‘standard’ /32 allocation from its local RIR. ISPs should consider this when making their address space requests to their regional registry.

5.4. Application Developers

5.4.1. Resolve addresses for each connection attempt

Applications should rarely cache DNS information, instead querying the resolver whenever DNS is required. This leaves the resolver to handle the Time-to-Live of the address records.

Application programmers should only cache DNS information if they are going to be opening a lot of connections in a short amount of time, and do not care about the possibilities of renumbering Ð this is up to the programmer’s judgement

5.4.2. Bind services to wildcard addresses INADDR_ANY, AF_UNSPEC

Application developers are encouraged to not bind to a specific label (e.g. from a configuration file), particularly as that label will likely only be resolved once (at invocation time) by a naming service.

Rather, it is recommended that all Internet-facing services are configured such that their server sockets bind to the unspecified address such that should new addresses become available for a node on which that service is installed and is desired to respond on, then incoming connection requests to that new address will be delivered by the stack to that service.

5.4.3. Promote SCTP as a stream-based connection-oriented transport protocol

For applications that require long-lived sessions and therefore desire to survive renumbering, mobility or other network episodes in which on-going connections would otherwise fail at the point where the old node address data was no longer valid, it is recommended that the Stream Control Transmission Protocol [RFC2960] be adopted in preference to TCP.

A key enabler of SCTP is its ability to dynamically change the endpoints associated with a live stream [SCTPDAR], meaning that in the stages where both old and new prefixes are available for a participant node, the association between sender and recipient can be extended to include the new address data; with the old data being removed at some point before the old address invalidates and becomes unusable.

5.5. Operating System and Router vendors

5.5.1. Expose DNS RR TTLs to applications

An extension to the Sockets API that exposes liveness metadata or DNS resource records (e.g. validity time of a resolved symbol at a minimum) would enable application and middleware developers to make a reasoned decision regarding the use of result data from the resolver library

5.5.2. Adhere to DNS RR TTLs

It is strongly recommended that resolver libraries strictly adhere to DNS Resource Record Time-to-Live data, and that they query for authoritative data when any cached copy has timed-out. This includes services such as name service caching applications.

Assuming that the network administrators at the site undergoing a renumbering episode are following the other recommendations in this memo, the authoritative data for the resource records should be such that the updated records should be picked up by all TTL-obeying resolvers in a timely manner.

5.5.3. Token-based addresses

An earlier recommendation to network designers discusses a feature of modern Solaris versions in which the interface identifiers for a node can be tokenised (i.e. specified explicitly without relying on SLAAC). Where router advertisements heard with prefix information blocks with the autonomous configuration bit set, that token is concatenated to the advertised network prefix to make a global address. This removes the need to maintain full hard-coded literals, and therefore aids in the renumbering process. This recommendation also applies, for example, to router vendors (e.g. Cisco IOS “IPv6 General Prefix”). It is recommended that other stack developers follow this lead and implement this much appreciated feature.

5.5.4. Invalidation of prefixes

Different OS vendors and versions currently exhibit inconsistent behaviour as regards the invalidation of SLAAC-enabled prefixes in various operating systems. Specifically, when router advertisements are received with a prefix validity time of zero (when already within the two hour minimum window), nodes should reset the validity time to two hours unless the advertisement is authenticated. It is recommended that operating system developers correctly implement prefix invalidation, particularly as it has an impact on the renumbering procedure.

5.5.5. Prefix invalidation without IPsec infrastructure

Further, where no IPsec infrastructure is available to authenticate router advertisements such that a zero validity lifetime is permitted by standards compliant nodes, the recommended practice for invalidating a prefix is thus: deprecate the prefix as usual (set preferred time to zero with a validity time no greater than two hours) and then when confident that all nodes are deprecated, remove the prefix from the advertisement set on the routers. Nodes will then count-down their validity time to zero and invalidate the prefix correctly.

5.6. Summary

This memo details a set of recommendations that ensue from work to date gaining operational experience with renumbering IPv6 networks. The recommendations are not exhaustive, particularly as there are many aspects of renumbering that impact on different scenarios in different ways.

The procedural recommendations made in this document have been shown to make renumbering IPv6 networks without a flag day to be achievable and a relatively trouble-free exercise.

Were vendors and developers to follow up on the recommendations targeted at them, the renumbering experience would be further simplified.

6. Conclusions and Future Work

This document has presented the results of enterprise-focused renumbering experiments undertaken at the Uiversities of Southampton and Muenster. The experiments have raised a number of issues, highlight for example some unpredicted host behaviour in the presence of multi-addressing during the [BAKER] procedure for IPv6 network renumbering.

The experiments have allowed a set of proposed recommendations to be created for further discussion and evaluation.

Renumbering tools for IPv6 networks will remain important while Provider Independent (PI) address space is difficult for eneterprise networks to obtain.

The results from D3.6.1 [6NET-D361] still hold; e.g. network management and monitoring tools need to be enhanced to better consider multi-addressed IPv6 nodes.

6.1. Future Work

A number of actions would be most helpful in advancing the understanding of the practical implications and robustness of IPv6 renumbering. We have examined some of the items from the list produced in D3.6.1, but some issues and tests remain. These include:

- An extended survey of the pervasiveness of address literals and steps to avoid their use
- Validation of IPv6 Prefix Delegation by DHCP, and of IPv6 Router Renumbering
- Better understanding of the commonalities and differences between renumbering and multi-homing (including new IPv6 shim6 WG proposals)
- Anecdotal experience of IETF participants or RIPE members that have undertaken an IPv6 renumbering exercise, e.g. in the transition from 3FFE::/16 6Bone addresses to production GAU addresses
- The use of ULA addresses in parallel with global addresses, to offer a stable addressing environment in the face of a renumbering event (but at what cost…?)
- New tools could be developed to assist the renumbering process, e.g. a Netflow-based tool to detect accesses to 'expiring' prefixes (we used a 'manual' version of this in the Southampton experiment described above).

The future work will be written up and where permitted by NDA will be openly published. The new 6DISS project [6DISS] is a likely primary dissemination channel.

As stated in D3.6.1, we argue in [THINK] that there may be a case to be made to reopen the PIER WG in the new context of IPv6, although that group has (apparently) not been active since 1997. This would require a BoF at a future IETF event. This would be best advanced if and when [THINK] is adopted as an IPv6 v6ops WG item.

The authors would be very happy to receive feedback and suggestions for improvements to the content of this cookbook, with a view to improving its usefulness and applicability, such that further point revisions can be issued beyond the lifetime of the 6NET project if necessary. Please send such comments to the editor, Tim Chown at *tjc@ecs.soton.ac.uk*.

In addition, readers are welcome to contact *helpdesk@6net.org* for specific assistance and guidance on issues to do with IPv6 network renumbering.

7. References

[6DISS] The 6DISS Project, http://www.6diss.org/

[6NET] The 6NET Project, http://www.6net.org/

[6NET-D361] “Cookbook for IPv6 network renumbering in SOHO and backbone networks”, 6NET Project Deliverable D3.6.1, http://www.6net.org/publications/deliverables/D3.6.1.pdf

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

[NAP] Velde, G., "IPv6 Network Architecture Protection", draft-ietf-v6ops-nap-00 (work in progress), March 2005.

[RFC2462] Thomson, S. and T. Narten, "IPv6 Stateless Address Autoconfiguration", RFC 2462 (under revision as draft-ietf-ipv6-rfc2462bis-08 under IESG evaluation), December 1998.

[RFC2960] Stewart, R. et al, “Stream Control Transmission Protocol”, RFC 2960, October 2000.

[RFC3177] IAB and IESG, "IAB/IESG Recommendations on IPv6 Address Allocations to Sites", RFC 3177, September 2001.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

[RFC3484] Draves, R., "Default Address Selection for Internet Protocol version 6 (IPv6)", RFC 3484, February 2003.

[RFC3513] Hinden, R. and S. Deering, "Internet Protocol Version 6 (IPv6) Addressing Architecture", RFC 3513, April 2003.

[RFC3531] Blanchet, M., “A Flexible Method for Managing the Assignment of Bits of an IPv6 Address Block”, RFC3531, April 2003.

[RFC3627] Savola, P., “Use of /127 Prefix Length Between Routers Considered Harmful”, RFC 3627, September 2003.

[RFC3633] Troan, O. and R. Droms, "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6", RFC 3633, December 2003.

[SCTPDAR] Stewart, R., "Stream Control Transmission Protocol (SCTP) Dynamic Address Reconfiguration", draft-ietf-tsvwg-addip-sctp-12 (work in progress), June 2005.

[THINK] Chown, T., Ford, A., Thompson, M. and S. Venaas, “Things to think about when Renumbering an IPv6 network”, draft-chown-v6ops-renumber-thinkabout-02 (work in progress), May 2005.

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

[V6OPS] IETF IPv6 Operations WG, http://www.ietf.org/html.charters/v6ops-charter.html

[VLAN-ID] Chown, T., “Use of VLANs for IPv4-IPv6 Coexistence in Enterprise Networks”, draft-chown-v6ops-vlan-usage-02 (work in progress), October 2004.

Appendix A: Address allocations

A candidate algorithm for the allocation of subnets under the new prefix in ECS is the Centremost method of RFC3531. That is:

1. The first round is to select only the middle bit (and if there is an even number of bits pick the bit following the center)
2. Create all combinations using the selected bits that haven't yet been created.
3. Start a new round by adding one more bit to the set. In even rounds add the preceding bit to the set. In odd rounds add the subsequent bit to the set.
4. Repeat 2 and 3 until there are no more bits to consider.

This gives rise to the first twenty candidate subnets being:

```
::000:0000::/64, ::020:0000::/64, ::040:0000::/64, ::060:0000::/64
::010:0000::/64, ::030:0000::/64, ::050:0000::/64, ::070:0000::/64
::080:0000::/64, ::0a0:0000::/64, ::0c0:0000::/64, ::0e0:0000::/64
::090:0000::/64, ::0d0:0000::/64, ::0f0:0000::/64, ::008:0000::/64
::028:0000::/64, ::048:0000::/64, ::068:0000::/64, ::018:0000::/64
```

Sequential but characterised

Another candidate, and the one deployed in the enactment of the renumbering exercise, reflects the characterisation of the ECS network as both an 'up-stream' for lower tiers and a direct provider to end user subnets.

Where address space is at a premium (i.e. for providing lower-tier topology), the strategy is to cluster aggregatable blocks together in the upper half of the enterprise's delegation, and thus the bottom /54 is used to provision end-user subnets.

The following Table shows how the 12-bits of the new ECS prefix have been partitioned.

	Binary bit pattern	Hex equivalent	Usage
3*MSBs 00	0000 0000 0000	000	Point-to-point links
	0000 0000 0001	001	First /64
	0000 0000 0010	002	Second /64
	…	…	…
	0011 1111 1110	3fe	1022nd /64
	0011 1111 1111	3ff	1023rd /64
3*MSBs 01	0100 0000 xxxx	400	First /60
	…	…	…

	0111 1111 xxxx	7f0	64th /60
8*MSB 1x	1000 xxxx xxxx	800	First /56
	1001 xxxx xxxx	900	reserved
	1010 xxxx xxxx	a00	Second /56
	1011 xxxx xxxx	b00	reserved
	1100 xxxx xxxx	c00	Third /56
	1101 xxxx xxxx	d00	reserved
	1110 xxxx xxxx	e00	Fourth /56
	1111 xxxx xxxx	f00	reserved

Appendix B: ECS Nameserver zone metadata

This shows the default time-to-live for ECS resource records, and includes an example of the label overloading for both multi-addressing, multi-protocol and load balancing (for the mail exchangers).

This zone file, completed with other resource data, was in-place after the MXes had their interfaces configured with addresses from both prefixes (autonomously).

Example BIND zone header:

```
$TTL 1800
@ 84600 IN SOA ns0.ecs.soton.ac.uk. hostmaster.ecs.soton.ac.uk. (
      2005061610 ; yyyymmddss **in GMT!!**
      7200 ; Refresh after 2 Hour
      3600 ; Retry after 1 Hour
      604800 ; Expire after 1 Week
      3600) ; Minimum TTL of 1 day
; NS records:
      84600 IN NS ns0.ecs.soton.ac.uk.
      84600 IN NS ns1.ecs.soton.ac.uk.
      84600 IN NS ns2.ecs.soton.ac.uk.
      84600 IN NS dns0.brad.ac.uk.
      84600 IN NS dns1.brad.ac.uk.
      84600 IN NS tortoise.webcentre.net.

ecs.soton.ac.uk. 300 IN MX 5 mx.ecs.soton.ac.uk.
ecs.soton.ac.uk. 300 IN MX 8 tortoise.webcentre.net.
mx.ecs.soton.ac.uk. 300 IN A 152.78.71.3 ; crow
mx.ecs.soton.ac.uk. 300 IN A 152.78.68.137 ; jackdaw
mx.ecs.soton.ac.uk. 300 IN A 152.78.71.63 ; bluetit
mx.ecs.soton.ac.uk. 300 IN AAAA 2001:630:d0:121::25 ; crow
mx.ecs.soton.ac.uk. 300 IN AAAA 2001:630:d0:115::25 ; jackdaw
mx.ecs.soton.ac.uk. 300 IN AAAA 2001:630:d0:121::26 ; bluetit
; 20050615 mkt - following three are due to renumbering exercise
mx.ecs.soton.ac.uk. 300 IN AAAA 2001:630:d0:f110::25 ; crow (RENO)
mx.ecs.soton.ac.uk. 300 IN AAAA 2001:630:d0:f102::25 ; jackdaw (RENO)
mx.ecs.soton.ac.uk. 300 IN AAAA 2001:630:d0:f110::26 ; bluetit (RENO)
```

Appendix C: Router Alerts

Current autonomously configured interface liveness data determined in various operating systems deployed in ECS:

Linux `ip show addr` in a shell

Win32 `show interface ipv6` in the `netsh` utility

FreeBSD `ifconfig -L` in a shell

OS/X as FreeBSD

This is an example configuration for rtadvd.conf on FreeBSD that shows a router primed for the introduction of a new prefix, 2001:cafe::/64, and the prefix information for the old prefix wound-down such that it is ready for deprecation.

Here is an example of prefix advertisement tuning in `rtadvd:`

```
default:\
      :chlim#64:raflags#0:rltime#1800:rtime#0:retrans#0:\
      :pinfoflags="la":vltime#604800:pltime#86400:mtu#0:
dc1:\
      :addr="2001:db8::":prefixlen#64:vltime#3600:pltime#1800:\
      :addr2="2001:cafe::":prefixlen2#64:vltime2#0:pltime2#0:\
      :pinfoflags2="l":
```

There is one key caveat when changing prefixes with rtadvd on FreeBSD: to change the preferred and validity times, the daemon needs to be restarted. Sending the TERM signal to the daemon (as suggested for graceful shutdown) will result in router advertisements being emitted with a Router Lifetime of zero, effectively telling all bound nodes that the router is not to be used as a default router. This is *not* desirable in the case where the operation is to add a prefix to the announcement set: the router still routes, and is a valid default router. Therefore, the recommendation is to send a KILL to the daemon (to shut it down) and then restart it with the new prefix configuration data. Whilst the daemon is down, hosts on-link will still direct packets to the router (which will route, for that function is unchanged), thus having minimal impact on local nodes.

Below is a similar configuration example for IOS:

```
interface FastEthernet 0/0
      ipv6 nd prefix-advertisement 2001:db8::/64 3600 1800 onlink \
           autoconfig
      ipv6 nd prefix-advertisement 2001:cafe::/64 0 0 onlink
```

Appendix D: Example packet filter issue

We have raised the issue of having local specific address data on both sides of a policy rule. Essentially, policies have to ensure that they catch all combinations of addresses whilst nodes and thus services are multi-addressed.

Using FreeBSD's `pf` as an example, the following configuration:

```
pass out quick on fxp0 proto tcp \
from 2001:db8:babe:1::/64\
to 2001:db8:babe:2::/64\
keep state
```

would require updating to be as follows:

```
pass out quick on fxp0 proto tcp \
from {2001:db8:babe:1::/64,2001:db8:beef:1::/64}\
to {2001:db8:babe:2::/64,2001:db8:beef:2::/64}\
keep state
```

Appendix E: Cisco IOS recommendation

Recent versions of Cisco IOS have an IPv6 general-prefix configuration feature that allows administrators to generalise their router configuration to achieve a tokenised address implementation. The following two examples demonstrate the feature's intended use. First, verbose IOS address configuration style:

```
interface FastEthernet1/0
     ipv6 address 2001:db8:beef::200:1/64

interface FastEthernet1/1
     ipv6 address 2001:db8:beef::300:1/64

interface FastEthernet2/0
     ipv6 address 2001:db8:beef::400:1/64
```

Then, IOS address configuration using an IPv6 General Prefix:

```
ipv6 general-prefix FOOTLE 2001:db8:beef::0/52

interface FastEthernet1/0
     ipv6 address FOOTLE ::200:1/64

interface FastEthernet1/1
     ipv6 address FOOTLE ::300:1/64

interface FastEthernet2/0
     ipv6 address FOOTLE ::400:1/64
```

However, whilst this is an excellent tool for assuring address consistency and easing prefix alteration, it is not without several drawbacks. Essentially, it should be the case that anywhere an address literal is used in an IOS configuration directive, a label-token pair should be permitted. This includes static routes, BGP summary configuration, access control lists, ipv6 prefix options, etc.

By way of example, the configuration snippet shown below would be an appropriate use of the general prefix feature, however it is not available in current IOS implementations, which renders the usefulness of the feature somewhat diminished.

```
ipv6 general-prefix FOOTLE 2001:db8:beef::0/52

interface FastEthernet1/0
     ipv6 address FOOTLE ::200:1/64
     ipv6 nd prefix FOOTLE ::200:0/64 no-autoconfig

interface Tunnel2
     ipv6 address FOOTLE ::1/112

ipv6 route FOOTLE 0:0:0:1000::/56 FOOTLE ::2
```

With a fully operational IPv6 general prefix feature, an extension that would assist renumbering exercises would be to include facility that multiple prefixes could be bound to the general-prefix label, with ‘stage’ markers indicating what particular status within a standardised renumbering procedure the router was in.

The stage marker would determine which of the prefixes bound to the label were ‘old’, which ‘new’, and therefore automatically associate different prefix options (lifetime, autoconf ability) according to the stage in the process.

The upshot of such a feature would be that the IOS configuration was still factored in that the number of configuration directives is minimised to cater for multiple prefixes during transition, and would still have all the benefits of minimising replicated data. Also, the control of the renumbering procedure would stay strictly in the grasp of the administrator: changing a router-wide configuration directive would impact on each set of interfaces, routes, etc. simultaneously.

Appendix F: Final address plan for ECS

The ECS Address plan under `2001:0630:00d0:f000::/52`, as of June 2005, is as follows:

New Prefix	Subnet Description
`000::/64`	Point-to-point links (/127s)
`001::/64`	6Core VLAN
`102::/64`	Servers VLAN
`103::/64`	IPv6 Test network
`104::/64`	DMZ VLAN
`105::/64`	Wireless DMZ VLAN
`106::/64`	DAB VLAN
`107::/64–10f::/64`	(free)
`110::/64`	Undergraduate Teaching VLAN
`111::/64`	IAM Group VLAN
`112::/64`	Systems Team VLAN
`113::/64`	ISIS Research Group VLAN
`114::/64`	Authenticated Wireless LAN VLAN
`115::/64`	(free)
`116::/64`	DSSE Research Group VLAN
`117::/64`	Printers VLAN (also labelled MMRG)
`118::/64–1ff::/64`	(free)
`200::/64`	SURGE (remote subnet)
`201::/64`	Chemistry (remote subnet)
`202::/64–3ff::/64`	(free)
`400::/60–7f0::/60`	Southampton Open Wireless Network
`800::/56`	OpenVPN remote connectivity provision
`900::/56`	(free)
`a00::/56`	AJF Tunnel Broker
`b00::/56`	(free)
`c00::/56`	Hexago Tunnel Broker
`d00::/56`	(free)
`e00::/56`	(free)
`f00::/56`	(free)