IPv6 Maintenance
Internet-Draft
Intended status: Informational
Expires: May 3, 2012

T. Chown, Ed.
University of Southampton
A. Matsumoto, Ed.
NTT

October 31, 2011

Considerations for IPv6 Address Selection Policy Changes
draft-ietf-6man-addr-select-considerations-04

Abstract

Where the source and/or destination node of an IPv6 communication is multi-addressed, a mechanism is required for the initiating node to select the most appropriate address pair for the communication. RFC 3484 (IPv6 Default Address Selection) [RFC3484] defines such a mechanism for nodes to perform source and destination address selection. While RFC3484 recognised the need for implementations to be able to change the policy table, it did not define how this could be achieved. Requirements have now emerged for administrators to be able to configure and potentially dynamically change RFC 3484 policy from a central control point, and for (nomadic) hosts to be able to obtain the policy for the network that they are currently attached to without manual user intervention. This text discusses considerations for such policy changes, including examples of cases where a change of policy is required, and the likely frequency of such policy changes. This text also includes some discussion on the need to also update RFC 3484, where default policies are currently defined.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

There have been various operational issues observed with Default Address Selection for IPv6 (RFC 3484) [RFC3484], as described in RFC 5220 [RFC5220]. As as a result, there has been some demand for hosts to be able to have their policy tables, and potentially the rules described in RFC 3484, modified dynamically. Such changes may apply to 'static' hosts in a network where policies or topologies change, or different default policy to that described in RFC 3484 is required, or for nomadic hosts within a network for which policies may vary depending on their location within the network.

2. Issues to Consider

There are a number of aspects to consider in the context of such address selection policy updates.

First is the frequency for which such updates are likely to be required; this can be determined largely from identifying the scenarios in which policy changes will be required. This may include overriding default operating system policies on startup, as well as changes while a system is running. We discuss this topic in Section 4.

Second, by understanding how dynamic the policy update mechanism needs to be we should be better placed to determine what types of update approaches best meet those needs. There may be other considerations of course, e.g. whether the systems are in managed or unmanaged environments, and whether the solution should be proactive or automated, as discussed in [I-D.ietf-6man-addr-select-sol]. Section 5 covers these issues.

Third, if we assume some policy update mechanism is defined we should consider how hosts and systems may become aware that a policy change has happened, and how policy can be disseminated in a timely fashion. Thus we need to understand what kind of triggers can be identified that can be used for invoking the policy table update mechanism, e.g. address re-obtainment, address lifetime expiration, or perhaps policy lifetime expiration. We also need to consider what other factors may come into play, e.g. potential policy conflicts. This is discussed in Section 6.

After analysing these issues, we can make some initial comments regarding the potential solution spaces, and what models may be well suited, e.g. push vs pull models, and what other methods might assist us, e.g. hints from local routing tables. This is covered in Section 7.

Finally, we should assess whether these update solutions require or need RFC 3484 to be updated. In some instances, we might envision solutions that simply use RFC 3484 as guidelines and provide sufficient controls to address the current limitations in the RFC. However, as noted in RFC 5220 [RFC5220], not all the operational issues observed to date can be remedied by updating RFC 3484 alone. There is already a good analysis of issues with RFC 3484 and how the text could be revised [I-D.ietf-6man-rfc3484-revise]).

3. Other Related Work

We note that there is some existing work in defining Requirements for Address Selection Mechanisms [RFC5221], and some initial work has been done in the solution space (for a DHCP-based method) [I-D.ietf-6man-addr-select-opt], but these are not discussed here. While RFC 5221 assumes that a dynamic policy update mechanism of some form is available, this draft is primarily aimed at understanding the scenarios and triggers for policy changes, to better inform future detailed solution discussions.

A draft discussing methods for multihoming without IPv6 NAT [I-D.ietf-v6ops-multihoming-without-nat66] has been published recently. This draft includes a requirement for a method to distribute address selection policy to support IPv6 multihoming.

4. Drivers for Policy Changes

If we wish to determine how frequent address selection policy changes are likely to be, we need to understand why such policies might need to be changed, for particular sites or networks.

One reference text for potential drivers for policy change is RFC 5220, in which operational issues with the existing policies described in RFC 3484 are listed. Each subsection of this document gives a reason why the existing rules or policy tables in RFC 3484 may not be sufficient in certain cases. There have been some significant changes to IPv6 since RFC 3484 was drafted which have impacted the RFC, e.g. the introduction of Unique Local Addresses (ULAs), and concerns about the impact of using longest prefix matching on (DNS) round-robin load balancing.

In summary, the issues raised in RFC 5220 were:

o Multiple Routers on a Single Interface

- o Ingress Filtering
- o Half-Closed Network Problem (*)
- o Combined Use of Global and ULA addresses (*)
- o Site Renumbering (*)
- o Multicast Source Address Selection (*)
- o Temporary Address Selection
- o IPv4 or IPv6 Prioritization (*)
- o ULA and IPv4 Dual-Stack Environment (*)
- o ULA or Global Prioritization (*)

The authors of RFC 5220 noted which of these issues can be solved just by changes to the RFC 3484 policy table, marked (*) above, and which cannot. It is interesting to note that issues largely related to internal networking and (administrative) policy decisions can be handled this way. However some issues need changes beyond just policy table updates.

4.1. Internal vs External Triggers

When considering drivers or triggers that may lead to a requirement for the policy to change, we can divide the problem space into those drivers that are external to a site or network and those internal to it. In the case of the first two examples above, a dynamic policy table update may be required by externally driven routing changes, assuming the site uses a dynamic routing protocol intra-site and the routing protocol is configured to reflect changes of extra-site routing topology.

If a site is multihomed using BGP and advertising a single prefix upstream, then no policy table manipulation is required for global address preferences. However where a site is multihomed by receiving a prefix from each upstream provider, each host will have multiple addresses and many need policy table manipulation. In such a case, the policy table of hosts may need to be updated according to the routing policy.

It should be noted that we have other mechanisms for dynamic routing topology change, for example deprecating one of the advertised prefixes, e.g. when one of the upstream links has a problem. But such mechanisms may only help in some cases, and do not remove the

need for agility in the RFC 3484 policy.

Other examples of external factors include a new transition mechanism being defined (e.g. as with the emergence of Teredo using 2001::/32 as assigned by IANA) and its inclusion being required in the policy table (at the time of writing Teredo is not included in RFC 3484, though some operating systems have added it), a new address block being defined, or a site renumbering event that could be triggered by an upstream provider's actions.

4.2. Administratively Triggered Changes

The other examples above are, in the general case, where the site administrator chooses to change a local policy and in doing so triggers the need for policy table updates. Some of these changes one might assume to be set once, and to change rarely, for example:

- o Setting priority use of IPv6 over IPv4 (or vice versa).
- o Setting priority use of ULAs over globals (or vice versa).
- o Setting priority of Teredo over native IPv4 (or vice versa).
- o Setting priority use of privacy addresses over DNS-published globals (or vice versa).
- o An internal network renumbering occurs, perhaps due to a site expanding.
- o The nature of the external connectivity through multiple ISPs requires specific additional information (policy) to be delivered to certain hosts (as discussed in 2.1.3 in RFC 5220).
- o Disabling longest-prefix match functions to facilitate round-robin load balancing.

However it may be the case that different parts of a site have different policies, or policies are changed in a rolling fashion across a site over time as IPv6 and/or ULAs are introduced (for example). This may happen where the administrator prefers a gradual introduction of new policy in a phased operation across a site, rather than changing policy across the whole site in one operation.

Other administrative changes may occur more frequently, e.g.:

- o Routing tables and forwarding tables change dynamically.

- o A different provider (link) is preferred for a given destination.

It's possible that provider links may vary on a daily basis, or by time of day. The frequency of such policy changes will depend on the frequency that the administrator wishes to change the implied traffic engineering policies.

4.3. Start-up vs Running Changes

When a host starts up it may be configured with the default RFC 3484 policies. At this stage a number of addresses may be configured on a number of interfaces on the host. At this time it may be desirable for the host to be able to receive the site-specific policy updates as a start-up override from the RFC 3484 defaults.

Other policy changes may later be required while the host is running. Ideally the same protocol should be used for the start-up and running state update mechanism.

4.4. Nomadic Nodes

A host may be nomadic within a site and as a result it may see the preferred policy change depending on the host's topological location within that site. Such a host should be capable of receiving policy updates in a timely fashion as it migrates within the network.

While this may be one case of 'running changes' described above, the policy changes are required due to the host's new point of attachment, not changes of policy to the current point of attachment. The frequency of updates are thus depend ant on the frequency of host mobility to parts of the network that have differing policies.

It is worth noting that the point at which a nomadic host configures its network settings would be an appropriate time for it to also receive any specific address selection policy for its point of attachement.

4.5. Multiple Interface Nodes

In considering scenarios where hosts may be multi-addressed and require policy to assist in address selection, the issue of hosts with multiple interfaces arises.

A host may have a variety of reasons to have multiple interfaces. It may for example have WiFi and 3G interfaces, and be capable of sending or receiving data over either interface. In some cases these interfaces may fall within the same administrative domain (ISP) and in some cases they may not. Another example would be the case of a

host with a VPN connection established, where address selection may be affected by the choice of whether the VPN connection is used or not. In this case it is interesting to note the choice to use the VPN tunnel for all, or just VPN home site traffic, is often left as a choice for the user via a tickbox selection. In addition, initiating the VPN typically changes several related settings, which is reasonable behaviour given the user chose to initiate the VPN connection.

Handling multiple interface nodes, and the possibility of conflicting policy being retrieved via each, is clearly an important problem today, but we note that RFC 3484 is currently defined as a per-node, not per-interface, mechanism (at least in the context of destination address selection). However, for RFC 3484, and its potential update mechanisms, to be applicable to typical 'real world' usage patterns, we should consider the multiple interface scenarios.

In the case where a host has multiple interfaces there are two likely scenarios:

- o Wired and wireless interfaces - in this case the operating system just needs to pick one interface and use it.

- o Normal and VPN interfaces - here the default should be the normal interface; the VPN interface should only be used for destinations associated with the VPN.

It has been suggested that an RFC 3484 policy table is required on a per-interface basis, though the choice of interface may itself be determined by the (destination) address selection process. As stated above, RFC 3484's policy table is currently defined to be node-wide. The node-wide problem is destination address selection when the source address is implied from a selected interface.

We note that there are some new, initial drafts published recently on the multiple interface problem [I-D.ietf-mif-problem-statement], and on a number of possible DHCPv6 extensions, e.g. to inform hosts about routing information to assist the selection process. [I-D.ietf-mif-dhcpv6-route-option], to inform hosts about DNS server selection policy, [I-D.ietf-mif-dns-server-selection]. These drafts fall within the remit of the new IETF mif WG. We note that the mif WG may produce relevant work with respect to the analysis of RFC 3484 policy changes, but at this stage no such output exists for inclusion.

5. How Dynamic?

The discussion above suggests that many of the potential triggers for policy table changes are 'one-off' in nature, i.e. a site makes a one-time policy change. It is thus unlikely that such administrative changes will be frequent.

There are some cases where updates may be required to be more frequent. In the example of a site which is implementing the gradual introduction of new policy across its network, while the frequency of changes may be relatively high, there is still probably only one or a small number of changes per host.

There may be a higher rate of policy changes within a site if there are nomadic hosts within the site, and these are roaming frequently to parts of the network where differing policies are in effect. In such cases it may be useful for a host to know whether or not the default RFC 3484 (or soon to be 3484bis) policies are in effect or not, and for there to be a 'cheap' way for the host to discover this.

Perhaps the biggest cause of policy change lies where the preferred links or paths for certain destinations change frequently over time as (typically) traffic engineering requirements change. In some networks this may be a daily change, or change between states at different times of day. It is not clear how common these cases are, and thus further input is welcomed here. Our belief is that cases where dynamic changes are used heavily are rare.

So, unless a site or network has rapidly changing traffic engineering requirements, or includes a high number of mobile nodes where the nodes are roaming to areas of the network with differing address selection related policies, the frequency of updates is likely to be relatively low. Most update requests will simply occur when a host starts up, and such requests for policy will be little different in frequency to other configuration requests. Other types of network change that may require a host to change its RFC 3484 policy behaviour are probably also likely to have associated changes with other host configuration data.

6. Considerations when Obtaining Policy

When a policy change is made, or a host migrates to a part of the network with different policies, that change of policy needs to be conveyed to the host. It needs to be made available and applied without restarting every affected host.

6.1. Changes in Available Address(es)

One might assume at first that when a host observes a change in its addresses, it should re-obtain the selection policy, but this may not always be the case. Not all policy changes are tied to a host changing one or more addresses, though it may be acceptable to query regardless for new policy (if a pull model is used) when address information changes.

As described above, it may be sufficient for a host to know when a policy is changed, or that perhaps the default policy is - or is not - in effect in its current locale.

6.2. Timeliness

In many, but not all, cases a policy change will need to be synchronised across a network. Thus there is a general issue of timely and synchronised dissemination of new policy. If the policy is distributed via the same mechanism that informs a host of a change of address(es), the application of the policy should be synchronised sufficiently with the address change. However, not all hosts may receive the update information at the same time, e.g. where new address assignments may be dependent on DHCP lease timers.

Where hosts use DHCPv6 for address information, in the absence of some form of Reconfigure message, a host may see a delay in policy changes being notified. One possible tool to help here is the DHCPv6 Lifetime Option (RFC4242) [RFC4242], which was originally introduced to assist with network renumbering events.

7. Solution Space

In this section we make some initial observations on the possible solution space.

7.1. Is default policy used?

There could be some mechanism to indicate to a host that the local network has a modified RFC 3484 policy in use, and thus that a revised policy table is available (and should be used). Alternatively a host could simply always attempt to obtain local RFC 3484 policy on startup. Regardless, it should also be possible for a host to detect that policy has changed (whether 'around' the host, or due to the host being nomadic). The method to convey this chnage to a host would depend on whether a push or pull configuration method is used.

It is assumed by 'default' policy here we refer to the revised/ updated RFC3484 specification, when that is produced.

7.2. Pull model

One potential solution is that a host uses a similar mechanism for RFC 3484 policy updates as is used for obtaining other configuration data, for example DHCPv6 [RFC3315]. For hosts using stateless autoconfiguration, policy could be made available via stateless DHCPv6 [RFC3736].

There are also already some initial proposals from the IETF mif WG on using DHCPv6 to deliver (mainly routing oriented) information to hosts, e.g. [I-D.ietf-mif-dhcpv6-route-option] and [I-D.ietf-mif-dns-server-selection]. These methods assume entities that have timely knowledge of routing information can provide equally timely hints to hosts on address selection, via DHCPv6. At this stage we believe that distributing RFC 3484 policy, as configured by an administrator, is a more practical use of DHCPv6.

The DHCP model allows individual nodes to potentially have differing policy, even when on the same subnet.

7.3. Push model

For hosts only using stateless autoconfiguration, in environments without stateless DHCPv6, it may be argued that since the network is not managed, there is not likely to be any managed policy to push to the hosts. In such environments hosts may perhaps more usefully use techniques such as router hints to make informed selections, as discussed later in this text.

It may of course be possible to piggy back policy information to a host in a Router Advertisement message, though initial consensus seems to be that this is a less attractive approach.

7.4. Routing Hints

As mentioned above, if a host has routing hints available, it may be able to make more informed selections. For example, a protocol could be specified for a node to query an on-link or remote (e.g. edge) router for 'hints'. For example, a new ICMPv6 message could be defined that queried a site edge router or route server for address pairs to use for a given destination address.

However, having hosts themselves participate in routing is generally not desirable. At this stage we can simply note that address selection might be simplified when some hint based on routing state

is provided to the end system, but such mechanisms are out of scope for this text.

It is noted in [RFC5887] that:

"In an environment where a site has more than one upstream link to the outside world, the site might have more than one valid routing prefix. In such cases, typically all valid routing prefixes within a site will have the same prefix length. Also in such cases, it might be desirable for hosts that obtain their addresses using DHCPv6 to learn about the availability of upstream links dynamically, by deducing from periodic IPv6 RA messages which routing prefixes are currently valid. This application seems possible within the IPv6 Neighbour Discovery architecture, but does not appear to be clearly specified anywhere."

The same thought seems relevant to address selection. There's no point selecting a source address whose prefix is not being advertised in RAs.

While routing and prefix hints may help a host make selection decisions, we should consider to what extent we wish to 'burden' a host with holding such information. If a host is to determine and cache routing hints, this may require an update of RFC 3484 policy table syntax to support preference for address pairs.

7.5. Policy Conflicts

In the case of a host operating in a single administrative domain, consistent policy should be available from whichever policy distribution mechanism provides the information. In such cases the network should not distribute policy sets from multiple entities (or by multiple mechanisms). However, in scenarios where a host is multi-addressed from multiple providers (e.g. a SOHO network with differing DSL and cable providers, or a user in a coffee shop initiating a VPN connection to their home network), multiple RFC 3484 policies may be received and there is likely to be some conflicts in the received policy information.

There are scenarios where a host may wish to ignore a conveyed policy. For example, the manager of a mobile node may not want to have its preferences changed by a visited network. In such a case one might argue that the mobile node should use MIPv6 with whatever its home network policies are.

The question then is whether the policy update mechanism itself needs to handle such potential conflicts, choosing one or ther other or merging by some set of heuristics, or whether the policy update

mechansism should be viewed independently of the conflict handling. The view of the design team was that distributing policy is a network problem, while handling conflicts is a host problem.

An initial draft on handling policy conflicts has been released separately [I-D.arifumi-6man-addr-select-conflict] given the topic is beyond the scope of this draft.

7.6. Policy Merging

For whatever mechanism is used to distribute RFC 3484 policy, it is not yet clear whether entire policy tables will be made available, or simply differences to the 'default', and thus whether policies may need to be merged, or overridden. Some policy conflicts will be unresolvable, e.g. one prefers IPv4 over IPv6, the other vice-versa. It may be simpler, though less efficient, for whole policy tables to be distributed, to avoid the merger problem.

One option may be to split the policy table into destination address selection and source address selection tables, with the policy distribution only updating the source address selection. Whether this might make merging policies simpler or in fact more complex would require further study.

It may also be possible to indicate some priority value for a policy, e.g. the priority of the interface it is received on, or perhaps to convey a unique identifier for the policy provider. Alternatibely, if there are multiple policies in conflict, a host could simply choose to fall back to use the default RFC 3484 policy.

A host also needs to know how to decide when to accept a policy. We could simplify the discussion by assuming a host is located in and only nomadic within a single site with one administrative controlling entity.

8. On RFC3484 Default Policies

RFC 3484 includes text about mechanisms for changing policy, having 'policy hooks' and having a configurable policy table. The implication is that defaults can be changed, and the text gives examples of this in Section 10. However, issues with RFC 3484 are broader that just policy table updates - it remains the case that some operational issues with RFC 3484 are not just related to the table, but on rules themselves, e.g. longest prefix match (affecting DNS round robin as described in [RFC5220]).

While discussing default policy, we noted that the word 'default' has

to be carefully defined, and also what the scope of this 'default' is. The default policy should be whatever RFC 3484, or its -bis version, states. At present some operating systems have already modified their default, based on operational feedback (e.g. on ULAs, on Teredo prefixes, or on the DNS round-robin problem). Currently we assume RFC3484 and changes to it will remain node-specific.

It certainly seems the case that the issues raised in RFC 5220, and discussed in [I-D.ietf-6man-rfc3484-revise] mean that an update of RFC 3484 is required, if only because some of the issues (as highlighted earlier) cannot be addressed by updating the policy table alone. An update would also give us some hope that all operating systems might have a common 'default'.

We do not note any specific comments here on how RFC 3484 should be updated. Other drafts have made suggestions. There are some discussions on ideas however, e.g. on the semantics of labels, and in adding ULAs explicitly to the default policy table.

There have also been new issues identified, e.g. on how one differentiates between IPv4+NAT access or IPv6 transitional access (e.g. via Teredo) to a dual-stack destination (the IPv4 private address inside the NAT is implicitly global, although its explicit scope is local) [I-D.denis-v6ops-nat-addrsel]. This illustrates that new issues may continue to be identified through growing IPv6 operational experience.

It is hard to predict exactly what features people will want to add to address selection algorithms in the future. Ideally we should not preclude future flexibility. It seems clear that any RFC 3484 update has two aspects: one that uses the existing policy table capability, and one that might change associated algorithms.

9. Conclusions

We believe a key outcome of this text should be progression of a solution to allow an enterprise network manager to configure their hosts with address selection policies that may differ from the RFC 3484 default, across all or part of their network, and possibly changing polciy with time. The general scope of this text applies to site and enterprise networks, where an administrator may need to change policies over time. It also includes nomadic nodes within the site, which may migrate to different parts of the site where different policies are required.

It is clear there may be environments which might introduce conflicting policies from different administrative domains, e.g. a

SOHO network with two ISP links, or an enterprise node running a VPN to a remote network. We conclude that the policy distribution mechanism is a network task, while policy conflict handling is a host task. Within this text, we do not present a solution for policy conflict handling, because at this time there is no perfect or practical solution. We thus recommend that we should progress the policy distribution solution while analysing conflict handling (which is not unique to this domain) in a separate text.

The scope of this text includes issues affecting the design of a protocol to allow a host's RFC 3484 policy table to be updated. From discussion of update triggers/scenarios, we believe rapid updates are unlikely to be required unless a node is in a network which has (very) dynamic external traffic engineering, or many nodes are mobile between parts of the network with differing policy. It's thus generally appropriate to use a similar method to obtain RFC 3484 policy as to obtain other configuration data.

In terms of obtaining policy, a pull-based solution, such as DHCPv6, may be more appropriate in managed environments (where managed non-default policies are most likely to be in effect), which would assure that hosts only gain policy information from a single entity (the DHCPv6 service). Use of DHCPv6 is also preferable if individual hosts on a subnet require different policies. In unmanaged networks, without stateless DHCPv6, use of routing hints may be an approach worth exploring.

Finally, there is a clear need to revise RFC 3484, to create a new default policy table for address selection, and to improve non policy table algorithms. This should be expedited.

10. Security Considerations

There are no extra Security consideration for this document.

11. IANA Considerations

There are no extra IANA consideration for this document.

12. Acknowledgements

The design team working on this draft is: Marcelo Bagnulo Braun, Marc Blanchet, Tim Chown, Francis Dupont, Tim Enos, TJ Evans, Brian Haberman, Tony Hain, Ruri Hiromi, Suresh Krishnan, Arifumi Matsumoto, Janos Mohacsi, Sebastien Roy, Teemu Savolainen, Fujisaki Tomohiro,

and John Zhao.

We also acknowledge comments received from IETF WG mail lists, including those by Brian Carpenter and Dave Thaler.

13. Informative References

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

[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.

[RFC3736] Droms, R., "Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6", RFC 3736, April 2004.

[RFC4242] Venaas, S., Chown, T., and B. Volz, "Information Refresh Time Option for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 4242, November 2005.

[RFC5220] Matsumoto, A., Fujisaki, T., Hiromi, R., and K. Kanayama, "Problem Statement for Default Address Selection in Multi-Prefix Environments: Operational Issues of RFC 3484 Default Rules", RFC 5220, July 2008.

[RFC5221] Matsumoto, A., Fujisaki, T., Hiromi, R., and K. Kanayama, "Requirements for Address Selection Mechanisms", RFC 5221, July 2008.

[RFC5887] Carpenter, B., Atkinson, R., and H. Flinck, "Renumbering Still Needs Work", RFC 5887, May 2010.

[I-D.ietf-6man-addr-select-sol]
Matsumoto, A., Fujisaki, T., and R. Hiromi, "Solution approaches for address-selection problems", draft-ietf-6man-addr-select-sol-03 (work in progress), March 2010.

[I-D.ietf-6man-rfc3484-revise]
Matsumoto, A., Kato, J., Fujisaki, T., and T. Chown, "Update to RFC 3484 Default Address Selection for IPv6", draft-ietf-6man-rfc3484-revise-04 (work in progress), July 2011.

[I-D.ietf-6man-addr-select-opt]
Matsumoto, A., Fujisaki, T., Kato, J., and T. Chown,

"Distributing Address Selection Policy using DHCPv6", draft-ietf-6man-addr-select-opt-01 (work in progress), June 2011.

[I-D.ietf-mif-problem-statement]
Blanchet, M. and P. Seite, "Multiple Interfaces and Provisioning Domains Problem Statement", draft-ietf-mif-problem-statement-15 (work in progress), May 2011.

[I-D.ietf-mif-dhcpv6-route-option]
Dec, W., Mrugalski, T., Sun, T., and B. Sarikaya, "DHCPv6 Route Options", draft-ietf-mif-dhcpv6-route-option-03 (work in progress), September 2011.

[I-D.ietf-mif-dns-server-selection]
Savolainen, T., Kato, J., and T. Lemon, "Improved DNS Server Selection for Multi-Interfaced Nodes", draft-ietf-mif-dns-server-selection-07 (work in progress), October 2011.

[I-D.arifumi-6man-addr-select-conflict]
Matsumoto, A., Fujisaki, T., and R. Hiromi, "Considerations of address selection policy conflicts", draft-arifumi-6man-addr-select-conflict-02 (work in progress), March 2010.

[I-D.denis-v6ops-nat-addrsel]
Denis-Courmont, R., "Problems with IPv6 source address selection and IPv4 NATs", draft-denis-v6ops-nat-addrsel-00 (work in progress), February 2009.

[I-D.ietf-v6ops-multihoming-without-nat66]
Troan, O., Miles, D., Matsushima, S., Okimoto, T., and D. Wing, "IPv6 Multihoming without Network Address Translation", draft-ietf-v6ops-multihoming-without-nat66-00 (work in progress), December 2010.

Authors' Addresses

Tim Chown (editor)
University of Southampton
Southampton, Hampshire SO17 1BJ
United Kingdom

Email: tjc@ecs.soton.ac.uk

Arifumi Matsumoto (editor)
NTT SI Lab
Midori-Cho 3-9-11
Musashino-shi, Tokyo 180-8585
Japan

Phone: +81 422 59 3334
Email: arifumi@nttv6.net