6MAN
Internet-Draft
Updates: 6724 (if approved)
Intended status: Standards Track
Expires: 8 January 2026

N. Buraglio
Energy Sciences Network
T. Chown
Jisc
J. Duncan
Tachyon Dynamics

7 July 2025

Prioritizing known-local IPv6 ULAs through address selection policy
draft-ietf-6man-rfc6724-update-23

Abstract

This document updates the default address selection algorithm for Internet Protocol Version 6 (IPv6), originally specified in RFC 6724, based on accumulated operational experience. It introduces the concept of "known-local" Unique Local Address (ULA) prefixes within the fd00::/8 block and specifies that ULA-to-ULA communications using such prefixes should be preferred over both IPv4-to-IPv4 and GUA-to-GUA (Global Unicast Address) communications in local use scenarios. The document defines mechanisms for nodes to identify and incorporate known-local prefixes into their address selection policy tables. It further clarifies the unconditional requirement for implementing Rule 5.5 of RFC 6724 and reduces the default precedence for 6to4 addresses. These updates enhance the supportability of typical deployment environments, including automatic and unmanaged configurations, and promote consistent IPv6-over-IPv4 precedence behavior for both ULA and GUA within local networks. The document acknowledges that certain atypical deployment models may require explicit configuration to achieve intended operational outcomes.

Table of Contents

1. Introduction

Since its publication in 2012, Default Address Selection for Internet Protocol Version 6 (IPv6) [RFC6724] has become an important mechanism by which nodes can perform address selection, deriving the most appropriate source and destination address pair to use from a candidate set by following the procedures defined in the RFC. Part of the process involves the use of a policy table, where the precedence and labels for address prefixes are listed, and for which a default policy table is defined.

It was always expected that the default policy table may need to be changed based on operational experience; section 2.1 of [RFC6724] states "It is important that implementations provide a way to change the default policies as more experience is gained" and points to the examples in Section 10 of the same document, which include Section 10.6 where a unique local address (ULA as defined in [RFC4193]) example is presented.

This document is written on the basis of such operational experience, in particular for scenarios where ULAs are used for their intended purpose as stated in [RFC4193], i.e., they are designed to be routed within a local site and by default not advertised, used or received from external locations to that site. The document defines how preference for ULAs may be elevated for appropriate, common scenarios.

To support the preference to use ULA address pairs over both IPv4 and GUA (Global Unicast Address as defined in [RFC3587]) address pairs for local intra-site scenarios, the concept of a "known-local" ULA address is introduced. This document describes the means for nodes to determine ULA prefixes that are known to be local to the site they are operating in and to insert those prefixes into their policy table with a label that differs from general ULA prefixes. This capability allows nodes to prefer ULA-ULA communication locally, but still use GUA-GUA address pairs for external communication, and importantly avoid selecting a ULA source to talk to a non-local ULA destination.

This document also reinforces the text in Section 5 of [RFC6724] to require support for Rule 5.5.

Section 3.1 of [RFC4193] defines ULAs within fc00::/7, where the L bit, as detailed in Section 3.1, is set to 1 for locally assigned (generated) prefixes, with L=0 as yet undefined. The use of known-locals as described in this document therefore applies to the currently used ULA prefixes under fd00::/8, where the prefixes conform to the definition in Section 3.1 of [RFC4193].

The overall goal of this update is to improve behavior for common scenarios, and to assist in the phasing out of use of IPv4, while noting that some specific scenarios may still require explicit configuration.

An IPv6 deployment, whether enterprise, residential or other, may use combinations of IPv6 GUAs, IPv6 ULAs, IPv4 global addresses, IPv4 RFC1918 addresses, and may or may not use some form of NAT. However, this document makes no comment or recommendation on how ULAs are used, or on the use of NAT in an IPv6 network.

1.1. Operational Issues Regarding Precedence for IPv4 addresses over ULAs

With multi-addressing being the norm for IPv6, more so where nodes are dual-stack, the ability for a node to pick an appropriate address pair for communication is very important.

Where getaddrinfo() as referenced in [RFC3493], or a comparable API is used, the sorting behavior should take into account both the source addresses of the requesting node as well as the destination addresses returned, and sort the candidate address pairs following the procedures defined in RFC6724.

The current default policy table leads to precedence for use of IPv6 GUAs over IPv4 global addresses, which is widely considered preferential behavior to support greater use of IPv6 in dual-stack environments. This helps in allowing sites to phase out IPv4 as its evidenced use becomes ever lower.

However, there are two issues with precedence, or rather non-precedence, for ULAs as originally defined in RFC6724.

First, the aforementioned default policy table places IPv6 ULAs below all IPv4 addresses, including [RFC1918] addresses, such that IPv4-IPv4 address pairs are favored over ULA-ULA address pairs. Given the IPv6 GUA preference, this could create difficulties with respect to planning, operational, and security implications for environments where ULA addresses are used in IPv4/IPv6 dual-stack network scenarios. The expected default prioritization of known-local IPv6 traffic over IPv4 by default, as happens with IPv6 GUA addresses, does not happen for ULAs.

As a result, the use of ULAs is not a viable option for dual-stack networking transition planning, large scale network modeling, network lab environments or other modes of large scale networking that run both IPv4 and IPv6 concurrently with the expectation that IPv6 will be preferred by default. Local preference of ULAs over IPv4 is thus important to assist administrators in phasing out IPv4 from dual-stack environments and is an important enabler for sites seeking to move from dual-stack to IPv6-only networking.

Additionally, an issue exists in the scenario where nodes in a dual-stack site are addressed from both ULA and GUA prefixes, RFC6724 will see GUA-GUA address pairs chosen over ULA-ULA. One goal of ULA addresses was to allow local communications to be independent of the availability of external connectivity and addresses, such that persistent ULAs can be used even when the global prefix made available to a site is withdrawn or changes.

This document therefore introduces two changes to RFC6724 to support a node implementing elevated or differential precedence for known-local ULAs, i.e., ULAs within a common local network, over both IPv4 and IPv6 GUAs.

The first change is an update to the default policy table to elevate the precedence for ULAs prefixes such that ULAs, like GUAs, carry a higher precedence than all IPv4 addresses, making IPv6 precedence over IPv4 consistent for both ULAs and GUAs.

The second change is the introduction of the concept of known-local ULAs. RFC6724 includes a method by which nodes may provide more fine-grained support for further elevating the preference for specific ULA prefixes, while leaving other general ULA prefixes at the precedence described in the previous paragraph. This document elevates the requirement for specific ULA prefixes to be inserted into the policy table to be a requirement, but only for observed prefixes that are known to be local, i.e., known-local ULAs.

These changes aim to improve the default handling of address selection for common cases, and unmanaged / automatic scenarios rather than those where DHCPv6 is deployed. The changes are discussed in more detail in the following sections, with a further section providing a summary of the proposed updates.

1.2. Precedence of 6to4 addresses

The anycast prefix for 6to4 relays was formally deprecated by [RFC7526] in 2015, and since that time the use of 6to4 addresses has further declined, with very little evidence of its use on the public Internet. Note that RFC7526 does not deprecate the 6to4 IPv6 prefix 2002::/16, it only deprecates the 6to4 Relay IPv4 prefix.

This document therefore demotes the precedence of the 6to4 prefix in the policy table to the same precedence as carried by the Teredo prefix defined in [RFC4380]. Leaving this entry in the default table will cause no problems and will help if any deployments still exist, and ensure 6to4 prefixes are differentiated from general GUAs.

The discussion regarding the adding of 6to4 site prefixes in section 10.7 of [RFC6724] remains valid.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

GUA: Global Unicast Addresses as defined in [RFC3587]

ULA: Unique Local Addresses as defined in [RFC4193]

Known-local ULA: A ULA prefix that an individual organization/site has determined to be local to a given node/network/administrative domain

RA: IPv6 Router Advertisement as defined in [RFC4861]

PIO: IPv6 Prefix Information Option as defined in [RFC4861]

SLAAC: IPv6 Stateless Address Auto-configuration [RFC4862]

3. Adjustments to RFC 6724

This document makes three specific changes to RFC6724: first to update the default policy table, second to change Rule 5.5 adjusts precedence of addresses in a prefix advertised by the next-hop to a requirement, and third to require nodes to insert observed known-local ULA prefixes into their policy table.

3.1. Policy Table Update

This update alters the default policy table listed in Rule 2.1 of RFC 6724.

It should be noted the order of rows in the policy table is of no consequence and only the precedence value is relevant.

The table below reflects the updated precedence table:

Prefix	Precedence	Label	
::1/128	50	0	
$known_local/48	45	14	(**)
::/0	40	1	
fc00::/7	30	13	(*)
::ffff:0:0/96	20	4	(*)
2002::/16	5	2	(*)
2001::/32	5	5	
::/96	1	3	
fec0::/10	1	11	
3ffe::/16	1	12	

(*) value(s) changed in update
(**) $known_local = the ULA Known-Local /48 IPv6 prefix(es) (if any) with precedence and labels per the rules in Sec 5.3

The update moves 2002::/16 to de-preference its status in line with [RFC7526] and moves the precedence of fc00::/7 above legacy IPv4, with ::ffff:0:0/96 now set to precedence 20.

3.2. Rule 5.5

The text in RFC6724 states that the Rules MUST be followed in order, but also includes a discussion note under Rule 5.5 that says that an IPv6 implementation is not required to remember which next-hops advertised which prefixes and thus that Rule 5.5 is only applicable to implementations that track this information.

This document removes that exception and elevates the requirement to prefer addresses in a prefix advertised by a next-hop router to a requirement for all nodes.

This change means that an IPv6 implementation will need to remember which next-hops advertised which prefixes [RFC8028], despite the conceptual models of IPv6 hosts in Section 5 of [RFC4861] and Section 3 of [RFC4191] having no such requirement.

3.3. Automatic insertion of known-local ULA prefixes into the policy table

Section 2.1 of [RFC6724] states that "an implementation MAY automatically add additional site-specific rows to the default table based on its configured addresses, such as for Unique Local Addresses (ULAs)", but it provides no detail on how such behavior might be implemented.

If a node can determine which ULA prefix(es) are known to be local, it can provide differential treatment for those over general, non-known-local ULAs, and insert these into the policy table at a higher precedence than GUAs while keeping all general ULA prefixes to a lower precedence.

This document thus elevates the MAY requirement above for insertion to a MUST for the specific case of known-local ULAs.

These known-local ULA prefixes are inferred from ULA addresses assigned to interfaces or learned from Prefix Information Options (PIOs) in Router Advertisements (RAs) [RFC4861] received on any interface regardless of how the PIO flags are set. Further, they are learned from Route Information Options (RIOs) in RAs received on any interface by Type C hosts that process RIOs, as defined in [RFC4191].

Section 3.1 of [RFC4193] only defines ULA prefixes where the L-bit is set to 1, i.e., prefixes under fd00::/8 where the prefix is locally assigned or generated.

The following rules define how the learnt known-local ULA prefixes under fd00::/8 are inserted into the address selection policy table for a node, through a conceptual list of known-local prefixes.

1. Any RIO or PIO that is delivered in an RA in which the "SNAC Router" RA header flag bit [SNACBIT] is set MUST be ignored when considering the following rules.

2. RIOs from within fd00::/8 are considered the preferred information source for determining known-local ULAs and should override other conflicting information or assumptions from other sources, including PIOs.

3. RIOs within fd00::/8 that are of length /40 or longer MUST be added to the known-local ULA list. RIOs for shorter prefixes MUST NOT be used to insert known-local ULA entries in the address selection policy table

4. PIOs received within fd00::/8 that are not already in the nodes known-local ULA list MUST be added to the list with an assumed prefix length of /48, regardless of how the PIO flags are set.

5. ULA interface addresses from within fd00::/8, particularly ones not created by SLAAC, and not already covered by the known-local ULA list MUST be added to the list with an assumed prefix length of /48. However, as with rule 1, if the ULA interface address was generated on the basis of a PIO that has only been seen in RAs in which the SNAC router flag bit is set, this ULA prefix MUST NOT be used as described in this rule (rule 5). This prevents potential use of a non-routable source address when communicating to a known-local ULA destination address that is not on the local link, as SNAC-generated ULAs can only work on a single link, and the only reason to ever choose them in source address selection is that the only choice for a destination address is the longest prefix match.

6. When inserting known-local ULA entries into the policy table, they MUST have a label of 14 (rather than the default ULA label of 13) and a precedence of 45.

7. Entries MUST be removed from the known-local ULA list and the Policy Table when the announced RIOs or PIOs are deprecated, or an interface address is removed, and there is no covering RIO or PIO.

When support is added for the insertion of known-local ULA prefixes into the current policy table it MUST default to on, but a mechanism SHOULD be supported to administratively toggle the behavior off and on.

Tools that display a node’s current policy table MUST show all currently inserted known-local ULA prefixes.

The identification and insertion of known-local prefixes under fc00::/8 is currently not defined.

Note that a practical limit exists on the number of RIOs and PIOs that can be placed into a single RA. Therefore, there is a practical limit to the number of known-local ULAs that can be expressed on a single network and the number of ULA prefixes that can automatically be preferred over IPv4 and GUA prefixes within the policy table. This limit is unlikely to impact most networks, especially residential and other small unmanaged networks that automatically generate ULA prefixes.

Section 4 of [RFC4191] says "Routers SHOULD NOT send more than 17 Route Information Options in Router Advertisements per link. This arbitrary bound is meant to reinforce that relatively few and carefully selected routes should be advertised to hosts." The exact limit will depend on other options that are used. So while this is not the practical limit discussed above, administrators should take extra care not to cause the RA size to exceed the MTU when filling the RA with RA Options when exceeding this limit.

Note that in the case of Rule 2 above it would be expected that ULA prefixes being included in the known-local prefix list be compliant with Section 3 of [RFC4193] (i.e., /48 in size) but the above rule is pragmatic in that it allows the use of ULA prefixes from /48 to /40 in length. Most networks use ("are expected to use") /48 prefixes as per RFC4193. However, it is possible that in some circumstances a larger managed enterprise may wish to use a shorter prefix (e.g., to simplify management, filtering rules, etc, and to overcome the issue with the number of RIOs an RA can carry as described in the above paragraph). However, such non-compliant use of ULAs may be problematic in other ways, e.g., carrying an increased risk of collision with other ULA prefixes, because shorter prefixes have a lower chance to be globally unique.

4. Configuration of the default policy table

As stated in Section 2.1 of [RFC6724] "IPv6 implementations SHOULD support configurable address selection via a mechanism at least as powerful as the policy tables defined here".

Based on operational experience to date, it is important that node policy tables can be changed once deployed to support future emerging use cases. This update thus re-states the importance of such configurability.

5. Intended behavior

In this section we review the intended default behavior after this update is applied.

5.1. GUA-GUA preferred over IPv4-IPv4

This is the current behavior, and remains unaltered. The rationale is to promote use of IPv6 GUAs in dual-stack environments.

5.2. GUA-GUA preferred over ULA-ULA

This is the current behavior, and remains unaltered for the general case.

However, where a ULA prefix is determined to be local, and added as a known-local ULA prefix to a node’s address selection policy table, communications to addresses in other known-local ULA prefixes will prefer ULA-ULA address pairs to GUA-GUA (matching label, higher precedence).

5.3. Known-local ULA - Known-local ULA preferred over GUA-GUA

As described in the previous case, this document elevates precedence for use of ULAs over GUAs in cases where the ULA prefix(es) in use can be determined to be local to a site or organization.

By only adapting this behavior for known-local ULAs, a node will not select a ULA source to talk to a non-local ULA destination and will instead correctly use GUA-GUA.

Nodes not yet implementing this RFC will continue to use GUA-GUA over ULA-ULA for all cases.

As an example, consider a site that uses prefixes ULA1::/48, ULA2::/48 and GUA1::/48.

Host A has address ULA1::1 and GUA1:1::1 Host B has address ULA2::1 and GUA1:2::1

Both ULA prefixes have been determined to be known-local through RIOs. Perhaps ULA2 is reachable within the site, but its prefix is not in direct use at host A.

If host A sends to host B the candidate pairs are ULA1::1 - ULA2::1 and GUA1:1::1 - GUA1:2::1.

In this case ULA1::1 - ULA2::1 wins because of matching labels (both 14) and higher precedence than GUA (45 vs 40).

If host A were to send to a host C with addresses ULA3::1 (where ULA3::/48 has not been learned to be a known-local prefix) and GUA2:1::1, host A would use the GUA address pair for the communication as the GUAs have matching labels (both 1) where the known-local ULA and general ULA do not (14 and 13 respectively).

5.4. Known-local ULA-ULA preferred over IPv4-IPv4

This update changes previous behavior for this case. RFC6724 as originally defined would lead to IPv4 being preferred over ULAs, which is contrary to the spirit of the IPv6 GUA precedence over IPv4, and to the goal of removing evidenced use of IPv4 in a dual-stack site before transitioning to IPv6-only.

This document elevates the precedence of known-local ULAs above IPv4, so known-local ULA-ULA address pairs will be chosen over IPv4-IPv4 pairs (matching label, higher precedence).

5.5. IPv4-IPv4 preferred over ULA-GUA

An IPv6 ULA address will only be preferred over an IPv4 address if both IPv6 ULA source and destination addresses are available. With Rule 5 of Section 6 of [RFC6724] and the ULA-specific label added in [RFC6724] (which was not present in [RFC3484]) an IPv4 source and destination will be preferred over an IPv6 ULA source and an IPv6 GUA destination address, even though generally known-local IPv6 ULA addresses are preferred over IPv4 in the policy table as proposed in this update. The IPv4 matching label trumps ULA-GUA.

6. Discussion of ULA source with GUA or remote ULA destination

In this section we present a discussion on the scenarios where a ULA source may be communicating with a GUA or ULA destination.

A potential problem exists when a ULA source attempts to communicate with GUA or remote ULA destinations. In these scenarios, the ULA source as stated earlier is by default intended for communication only with the local network, meaning an individual site, several sites that are part of the same organization, or multiple sites across cooperating organizations, as detailed in [RFC4193]. As a result, most GUA and ULA destinations are not attached to the same local network as the ULA source and are, therefore, not reachable from the ULA source.

Scenario 1: ULA source and GUA destination

When only a ULA source is available for communication with GUA destinations, this generally implies no connectivity to the IPv6 Internet is available. Otherwise, a GUA source would have been made available and selected for use with GUA destinations. As a result, the ULA source will typically fail when it attempts to communicate with most GUA destinations. However, corner cases exist where the ULA source will not fail, such as when GUA destinations are attached to the same local network as the ULA source.

Scenario 2: ULA source and remote ULA destination

Receiving a DNS response for a ULA destination that is not attached to the local network is considered a misconfiguration. This contradicts the operational guidelines provided in Section 4.4 of [RFC4193]. Nevertheless, this can occur, and the ULA source will typically fail when it attempts to communicate with ULA destinations that are not attached to the same local network as the ULA source. This case provides a rationale for implementing support for known-local ULA prefix insertion in the policy table, such that differential behavior can be applied for known-local versus general ULA prefixes.

The remainder of this section discusses several complementary mechanisms involved with these scenarios.

6.1. The ULA Label and its Precedence

[RFC6724] added (in obsoleting [RFC3484]) a separate label for ULAs (the whole range, under fc00::/7), whose default precedence is raised by this update. This separate label interacts with Rule 5 of Section 6 of [RFC6724], which says:

Rule 5: Prefer matching label.

If Label(Source(DA)) = Label(DA) and Label(Source(DB)) <> Label(DB), then prefer DA.

Similarly, if Label(Source(DA)) <> Label(DA) and Label(Source(DB)) = Label(DB), then prefer DB.

In the first scenario, the ULA source label, whether known-local or not, will not match the GUA destination label. Therefore, an IPv4 destination, if available, will be preferred over a GUA destination with a ULA source, even though the GUA destination has higher precedence than the IPv4 destination in the policy table. This means the IPv4 destination will be moved up in the list of destinations over the GUA destination with the ULA source.

If the ULA (fc00::/7) label is removed from the policy table, a GUA destination with a ULA source will be preferred over an IPv4 destination, as GUA and ULA will be part of the same label (for ::/0).

In the second scenario, if the ULA source has been recognized as being within a known-local prefix that has been inserted into the address selection policy table, then the known-local ULA source and general ULA destination will have different labels, and therefore IPv4 communication will be preferred.

If the ULA source has not been recognized as known-local, e.g., if the insertion of known-local prefixes into the policy table has been administratively disabled, its general ULA label will match the general ULA destination label and therefore, whether part of the local network or not, the ULA destination will be preferred over an IPv4 destination.

6.2. Happy Eyeballs

Regardless of the precedence resulting from the above discussion, Happy Eyeballs version 1 [RFC6555] or version 2 [RFC8305], if implemented, will try both the GUA or ULA destination with the ULA source and the IPv4 destination and source pairings. The ULA source will typically fail to communicate with most GUA or remote ULA destinations, and IPv4 will be preferred if IPv4 connectivity is available unless the GUA or ULA destinations are attached to the same local network as the ULA source.

6.3. Try the Next Address

As stated in Section 2 of [RFC6724]:

"Well-behaved applications SHOULD NOT simply use the first address returned from an API such as getaddrinfo() and then give up if it fails. For many applications, it is appropriate to iterate through the list of addresses returned from getaddrinfo() until a working address is found. For other applications, it might be appropriate to try multiple addresses in parallel (e.g., with some small delay in between) and use the first one to succeed."

Therefore, when an IPv4 destination is preferred over GUA or ULA destinations, IPv4 will likely succeed if IPv4 connectivity is available, and the GUA or ULA destination may only be tried if Happy Eyeballs is implemented.

On the other hand, if the GUA or ULA destination with the ULA source is preferred, the ULA source will typically fail to communicate with GUA or ULA destinations that are not connected to the same local network. However, if the operational guidelines in Section 4.3 of [RFC4193] are followed, recognizing this failure can be accelerated, and transport layer timeouts (e.g., TCP hard errors as described in section 2.1 [RFC5461]) can be avoided. The guidelines will cause a Destination Unreachable ICMPv6 Error to be received by the source device, signaling the next address in the list to be tried, as discussed above.

7. Following ULA operational guidelines in RFC4193

This section re-emphasizes two important operational requirements stated in [RFC4193] that should be followed by administrators.

7.1. Filtering ULA-source addresses at site borders

Section 4.3 of [RFC4193] states "Site border routers and firewalls should be configured to not forward any packets with Local IPv6 source or destination addresses outside the site, unless they have been explicitly configured with routing information about specific /48 or longer Local IPv6 prefixes".

And further that "Site border routers should respond with the appropriate ICMPv6 Destination Unreachable message to inform the source that the packet was not forwarded".

As stated in the above discussion, such ICMPv6 messages can assist in fast failover for TCP connections.

7.2. Avoid using ULA addresses in the global DNS

Section 4.4 of [RFC4193] states that "AAAA and PTR records for locally assigned local IPv6 addresses are not recommended being installed in the global DNS."

This is particularly important given the general method presented in this document elevates the priority for ULAs above IPv4. However, where support for insertion of known-local prefixes is implemented, such "rogue" ULAs in the global DNS are a less serious concern for address selection as they would have the lowest precedence.

8. The practicalities of implementing address selection support

As with most adjustments to standards, and using the introduction of RFC6724 as a measuring stick, the updates defined in this document will likely take several years to become common enough for consistent behavior within most operating systems. At the time of writing, it has been over 10 years since RFC6724 has been published but we continue to see existing commercial and open source operating systems exhibiting RFC3484 (or other) behavior.

While it should be noted that RFC6724 defines a solution to adjust the address precedence selection table that is functional theoretically, operationally the solution is operating system dependent and in practice policy table changes cannot be signaled by any currently deployed network mechanism. While [RFC7078] defines such a DHCPv6 option, there are few if any implementations. This lack of an intra-protocol or network-based ability to adjust address selection precedence, along with the inability to adjust a notable number of operating systems either programmatically or manually, renders operational scalability of such a mechanism challenging.

It is especially important to note this behavior in the long lifecycle equipment that exists in industrial control and operational technology environments due to their very long mean time to replacement/lifecycle.

9. Limitations of RFC6724

The procedures defined in RFC6724 do not give optimal results for all scenarios. As stated in the introduction, the aim of this update is to improve the behavior for the most common scenarios.

Operational experienced has demonstrated that 3484/6724/getaddrinfo() model is fundamentally limited with regard to optimal address selection. A model that considers address pairs directly, rather than sorting on destination addresses with the best source for that address, would be preferable, but beyond the scope of this document.

To simplify address selection, administrators may instead look to deploy IPv6-only and/or may choose to only use GUA addresses and no ULA addresses. Other approaches to reduce the use of IPv4, e.g., through use of DHCPv4 Option 108 as defined in [RFC8925] as part of an "IPv6 Mostly" deployment model, also help simplify address selection for nodes.

10. Acknowledgements

The authors would like to acknowledge the valuable input and contributions of the 6man WG including (in alphabetic order) Erik Auerswald, Dale Carder, Brian Carpenter, Tom Coffeen, Lorenzo Colitti, Chris Cummings, David Farmer (in particular for the ULA to GUA/ULA discussion text, and discussion of using the specific fd00::/8 prefix for known-locals), Bob Hinden, Scott Hogg, Ed Horley, Ted Lemon, Jen Linkova, Michael Richardson, Kyle Rose, Nathan Sherrard, Ole Troan, Eduard Vasilenko, Eric Vyncke, Paul Wefel, Timothy Winters, and XiPeng Xiao.

11. Implementation Status

This section should be removed before publication as an RFC.

There are two known implementations of the ULA known-local precedence mechanism. The first implementation was created by Lorenzo Colitti at Google as a prototype solution, with public code available for reference on their android platform available to the public [ANDROID]. It was last updated in April of 2024, and does not include the capability to listen for RIO/PIO changes, but does support adding the ULA prefix learned on the interface to the known-local precedence.

The second implementation was written by Jeremy Duncan at Tachyon Dynamics and made available as open source, reference prototype code available [RAIO-ULA-PY]. This implementation includes a full implementation written in python, including the capability to listen to RIO and PIO on the wire and adjust ULA known-local prefixes as needed. It was last updated in May of 2024.

12. Security Considerations

The mixed precedence for IPv6 over IPv4 from the default policy table in RF 6724 represents a potential security issue, given an operator may expect ULAs to be used when in practice RFC1918 addresses are used instead.

The requirements of RFC4193, stated earlier in this document, should be followed for optimal behavior.

Administrators should be mindful of cases where communicating nodes have differing behavior for address selection, e.g., RFC3484 behavior, RFC6724, the updated RFC6724 behavior defined here, some other non-IETF-standardized behavior, or even no mechanism. There may thus be inconsistent behavior for communications initiated in each direction between two nodes. Ultimately all nodes should be made compliant to the updated specification described in this document.

13. IANA Considerations

None.

14. Appendix

The table below reflects the [RFC6724] table

	RFC6724	
Prefix	Precedence	Label
::1/128	50	0
::/0	40	1
::ffff:0:0/96	35	4
2002::/16	30	2
2001::/32	5	5
fc00::/7	3	13
::/96	1	3
fec0::/10	1	11
3ffe::/16	1	12

15. Summary of changes and additional text since RFC6724

* Introduced concept of known-locals and rules for their insertion/ removal in the table.

* Changed default policy table to move fc00::/7 to precedence 30, above legacy IPv4.

* Changed default policy table to move the 6to4 address block 2002::/16 to the same precedence as the Teredo prefix.

* Changed ::ffff:0:0/96 to precedence 20.

* Changed Rule 5.5 to a MUST support.

* Added text clarifying intended behavior.

* Added text discussing ULA to GUA/ULA case.

* Added text for the security section.

* Added text to account for SNAC bit.

16. References

16.1. Normative References

[RFC6724] Thaler, D., Ed., Draves, R., Matsumoto, A., and T. Chown, "Default Address Selection for Internet Protocol Version 6 (IPv6)", RFC 6724, DOI 10.17487/RFC6724, September 2012, <https://www.rfc-editor.org/info/rfc6724>.

[RFC8028] Baker, F. and B. Carpenter, "First-Hop Router Selection by Hosts in a Multi-Prefix Network", RFC 8028, DOI 10.17487/RFC8028, November 2016, <https://www.rfc-editor.org/info/rfc8028>.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, DOI 10.17487/RFC4861, September 2007, <https://www.rfc-editor.org/info/rfc4861>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC4191] Draves, R. and D. Thaler, "Default Router Preferences and More-Specific Routes", RFC 4191, DOI 10.17487/RFC4191, November 2005, <https://www.rfc-editor.org/info/rfc4191>.

[RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast Addresses", RFC 4193, DOI 10.17487/RFC4193, October 2005, <https://www.rfc-editor.org/info/rfc4193>.

[RFC7526] Troan, O. and B. Carpenter, Ed., "Deprecating the Anycast Prefix for 6to4 Relay Routers", BCP 196, RFC 7526, DOI 10.17487/RFC7526, May 2015, <https://www.rfc-editor.org/info/rfc7526>.

[SNACBIT] "SNAC Router Flag in ICMPv6 Router Advertisement Messages", n.d., <https://datatracker.ietf.org/doc/draft-ietf-6man-snac-router-ra-flag/>.

[ANDROID] "Optionally prefer known-local ULAs in Android", n.d., <https://r.android.com/3046000>.

[RAIO-ULA-PY]
"Python known-local ULA implementation", n.d.,
<https://github.com/jeremy-duncan/raio_ula>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

16.2. Informative References

[RFC1918] Rekhter, Y., Moskowitz, B., Karrenberg, D., de Groot, G. J., and E. Lear, "Address Allocation for Private Internets", BCP 5, RFC 1918, DOI 10.17487/RFC1918, February 1996, <https://www.rfc-editor.org/info/rfc1918>.

[RFC6555] Wing, D. and A. Yourtchenko, "Happy Eyeballs: Success with Dual-Stack Hosts", RFC 6555, DOI 10.17487/RFC6555, April 2012, <https://www.rfc-editor.org/info/rfc6555>.

[RFC8305] Schinazi, D. and T. Pauly, "Happy Eyeballs Version 2: Better Connectivity Using Concurrency", RFC 8305, DOI 10.17487/RFC8305, December 2017, <https://www.rfc-editor.org/info/rfc8305>.

[RFC3587] Hinden, R., Deering, S., and E. Nordmark, "IPv6 Global Unicast Address Format", RFC 3587, DOI 10.17487/RFC3587, August 2003, <https://www.rfc-editor.org/info/rfc3587>.

[RFC8925] Colitti, L., Linkova, J., Richardson, M., and T. Mrugalski, "IPv6-Only Preferred Option for DHCPv4", RFC 8925, DOI 10.17487/RFC8925, October 2020, <https://www.rfc-editor.org/info/rfc8925>.

[RFC3484] Draves, R., "Default Address Selection for Internet Protocol version 6 (IPv6)", RFC 3484, DOI 10.17487/RFC3484, February 2003, <https://www.rfc-editor.org/info/rfc3484>.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, DOI 10.17487/RFC4862, September 2007, <https://www.rfc-editor.org/info/rfc4862>.

[RFC3493] Gilligan, R., Thomson, S., Bound, J., McCann, J., and W. Stevens, "Basic Socket Interface Extensions for IPv6", RFC 3493, DOI 10.17487/RFC3493, February 2003, <https://www.rfc-editor.org/info/rfc3493>.

[RFC4380] Huitema, C., "Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs)", RFC 4380, DOI 10.17487/RFC4380, February 2006, <https://www.rfc-editor.org/info/rfc4380>.

[RFC5461] Gont, F., "TCP's Reaction to Soft Errors", RFC 5461, DOI 10.17487/RFC5461, February 2009, <https://www.rfc-editor.org/info/rfc5461>.

[RFC7078] Matsumoto, A., Fujisaki, T., and T. Chown, "Distributing Address Selection Policy Using DHCPv6", RFC 7078, DOI 10.17487/RFC7078, January 2014, <https://www.rfc-editor.org/info/rfc7078>.

Authors' Addresses

Nick Buraglio
Energy Sciences Network
Email: buraglio@forwardingplane.net

Tim Chown
Jisc
Email: Tim.Chown@jisc.ac.uk

Jeremy Duncan
Tachyon Dynamics
Email: jduncan@tachyondynamics.com