6MAN Working Group
Internet-Draft
Updates: RFC2464, RFC4291, RFC4861, RFC4862, RFC7136, RFC8273 (if approved)
Intended status: Standards Track
Expires: May 3, 2021

G. Mishra
Verizon Inc.
A. Petrescu
CEA, LIST
N. Kottapalli
Benu Networks
N. Kottapalli
Ciena
D. Shytyi
SFR

October 30, 2020

# SLAAC with prefixes of arbitrary length in PIO (Variable SLAAC) draft-mishra-6man-variable-slaac-01

Abstract

This draft proposes the use of arbitrary length prefixes in PIO for SLAAC. A prefix of length 65 in PIO, for example, would be permitted to form an addresses whose interface identifier length is length 63, which allows several benefits.

In the past, various IPv6 addressing models have been proposed based on a subnet hierarchy embedding a 64-bit prefix. The last remnant of IPv6 classful addressing is a inflexible interface identifier boundary at /64. This document proposes flexibility to the fixed position of that boundary for interface addressing.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on May 3, 2021.

Table of Contents

## 1. 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.

## 2. Introduction

From the beginning, the IPv6 addressing plan was based on a 128 bit address format made up of 8 hextets which were broken down into a 64 bit four hextet prefix and 64 bit four hextet interface identifier. For example, the address 2001:db8:3:4::1 has the first 4 hextets forming the /64 prefix 2001:db8:3:4::/64, whereas the last four hextets form an interface identifier abbreviated as ::1 (a 'hextet' is a group of max 4 hex digits between two columns, e.g. "2001" and "db8" are each a hextet). A comprehensive analysis of the 64-bit boundary is provided in [RFC7421]. The history of IPv6 Classful models proposed, and the last remnant of IPv6 Classful addressing rigid network interface identifier boundary at /64 is discussed in detail as well as the removal of the fixed position of the boundary for interface addressing in draft [I-D.bourbaki-6man-classless-ipv6].

This document discusses the reasons why the interface identifier has been fixed at 64 bits, and the problems that can be addressed by changing the GUA interface identifier from fixed 64 bit size to a variable interface identifier. This change would be consistent with static and DHCPv6 stateful IPv6 address assignment, as well as the proposed solution would ensure maintaining backwards compatibility for existing implementations. This document tries to achieve clearing the confusion related to prefix length, and provide consistency of variable length prefix across the three IPv6 addressing strategies deployed, static, DHCPv6 and Stateless Address Autoconfiguration(SLAAC), and finally update all RFCs with the new variable SLAAC standard. The 64 bit fixed boundary problem statment is defined in draft [I-D.mishra-v6ops-variable-slaac-problem-stmt].

Over the years one of the merits of increasing the prefix length, and reducing the size of the interface identifier has been incorrectly stated as the possibility of IPv6 address space exhaustion could be

circumvented, or that a 64 bit interface identifier is a wasteful use of address space.

## 3. The History behind the 64 bit fixed boundary

The fixed length of an Interface Identifier has roots in other early non-IP networks such as IPX of Novell and another from Apple.

Over the course of the history of the IPv6 protocol, several addressing models have been proposed to break up the prefix into a hierarchical format. One of the first attempts was [RFC2450] which was based on a 13 bit Level Aggregation (TLA), 24 bit Next-Level Aggregation (NLA), 16 bit Site Level aggregator Identifiers. The current IPv6 addressing architecture for global unicast addressing uses [RFC3587] for global unicast address currently being delegated by IANA 2000::/3 prefix. With the recommendation in [RFC3177] which called for a default end site assignment of a /48 which was adopted by the Regional Internet Registry was revised with [RFC6177] to a smaller block size of /56 prefix to end sites to avoid risk of premature address depletion. The current IPv6 addressing architecture [RFC3587] for global unicast addressing was now based on an IPv6 hierarchical format which now consists of a 45 bit global routing prefix, 16 bit subnet ID followed by 64 bit interface identifier. In the earlier deployments of IPv6 due to the stringent guidelines of [RFC4291] which stated that for all unicast addresses, except those that start with the binary value 000, Interface IDs are required to be 64 bits long and to be constructed in Modified EUI-64 format. Referencing IPv6 Addressing architecture [RFC3513] section 2.5.5 depicts examples of global unicast addresses that start with binary 000 are IPv6 addresses with embedded IPv4 addresses and IPv6 address containing encoded NSAP addresses [RFC4548] described in [RFC6052]. An example use case would be for NAT64 [RFC6146] as well as many other use cases that exist with transition technology tunneling using IPv4 IPv6 translators.

The general format for IPv6 global unicast addresses is as follows:

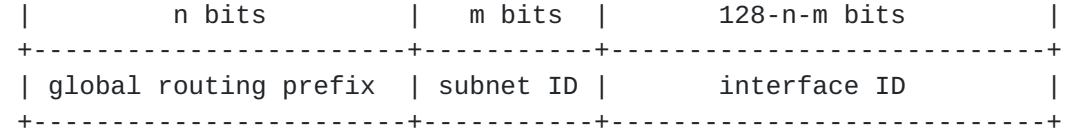


Figure 1: Format of the IPv6 global unicast addresses

Even though [RFC4291] states that all global unicast addresses except those that start with binary value 000, which use ipv4 ipv6

translators [RFC6052], that static and DHCPv6 violates [RFC4291] as variable length masking to 128 is supported, where SLAAC variable length masking remains forbidden. IPv6 packets over LAN based technologies such as ethernet must use 64 bit interface identifier per [RFC2464]. Nothing is mentioned regarding wireless based technologies such as MIP6, V2V or 6loWPAN, with regards to interface identifier length stringent requirement for 64 bit prefix length. Stateful Address Autoconfiguration [RFC4862] states that the sum total of the prefix length and interface identifier should equal 128 bits, but does not state that the interface identifier should be 64 bits. Note that [RFC4861] states that the PIO (Prefix information Options), that the A-bit Autonomous address-configuration flag when set indicates that the prefix can be used for (SLAAC) stateless address autoconfiguration, and [RFC4862] states to silently ignore the PIO options if the A flag is not set in the Router Advertisement. If the A flag is not set then /64 is only a recommendation which applies to DHCPv6 and static.

During the early deployments of IPv6, /64 was a 'de facto' standard prefix length for deployment to all router interfaces including point-to-point and loopbacks. In early deployments of IPv6, due to the complexity and overall learning curve, and change going from IPv4 to IPv6, the keep it simple approach of /64 everywhere was the general rule of thumb for deployment. After decades of deployment, operators started to dig further into how IPv4 started out as classful with classful routing protocols such as RIP or IGRP. Later as Classless inter-domain routing with BGP became mainstream with larger enterprises and service providers, operators started looking at IPv6 and variable length masking. Operators now started experimenting trying to subnet at nibble boundaries to start and became brave enough to tackle subnetting on a bit boundary. As variable length subnet masking became more mainstream with IPv6, operators started to use /126 mask on point-to-point links. Around that time [RFC3627] came out which talked about the harmful effects of /127 and that it was forbidden due to operational impacts. Harmful impacts of /127 were due to subnet-router anycast being in conflict with [RFC2526] /121. Later was found the benefits of /127 avoided the ping-pong effect and the subnet-router anycast conflict could be avoided by disabling Duplicate address detection thus the status of use of /127 on point-to-point links was updated by [RFC6164]. As the evolution of IPv6 continued, questions would come as to why the interface identifier is so large at 64 bits, as 64 bits equates to 18,446,744,073,709,551,616 IPv6 addresses, which is more than anyone could ever imagine on a single flat subnet far into the distant future. The main reason for the larger 64 bit interface identifier is for privacy when connected directly to the internet, or on an unsecure public hotspot or location so your device is not traceable.

From the beginning of IPv6 deployments most enterprises went with SLAAC, but as DHCPv6 matured, enterprises migrated to DHCPv6, and network infrastructure remained configured manually using static configurations. Since so many RFC's mention the SLAAC 64 bit boundary requirement and confusion related to this topic, in fact prevented operators proliferation of even attempting to use longer prefixes on host subnets with static or DHCPv6 stateful. Most IPv6 implementations even to this day do not use longer than 64 bit prefixes, and still maintain the 64 bit boundary for host subnet, for both DHCPv6 and static, even though technically feasible, due to fear of interoperability issues that may arise. With this new evolution of IPv6 addressing architecture with variable SLAAC, we can bring back SLAAC to the mainstream for all IPv6 deployments. This will also allow operators to now comfortably deploy both DHCPv6 and static with greater than 64 bit prefix length to host subnets, without fear of interoperability problems.

Today we have three methods of IPv6 address deployment, SLAAC, DHCPv6 and static. DHCPv6 does not provide an adequate IPv6 addressing solution as described in detail in the DHCPv6, Static, and SLAAC comparison section. As user subnets flatten out further, as the IPv4 under pinning is eliminated, removing the shackles on IPv6, the subnets will get much flatter. As the subnets flatten out in large Enterprise networks where you have 100's of Dual Stack subnets migrate to a single "IPV6-ONLY" subnet, the overhead DHCPv6 Normal mode messaging becomes exacerbated. The problem with DHCPv6 is that once the "M" managed bit is set to "1", all hosts on the subnet cache the M bit and change to DHCPv6 stateful mode. Higher probability of rouge devices such as printers or other appliances misbehaving with IPv6 enabled by default, now in DHCPv6 mode, spewing of millions of DHCPv6 messages that can now impact the router control plane processing of packets. This can be alleviated with special custom Control Plane policer policy, however now adds complexity and administrative overhead to DHCPv6 deployments. Enterprises and Service Providers require a viable IPv6 deployment solution that can accommodate the shortfalls of both static and DHCPv6 addressing. Static addressing due to administrative overhead of manual assignment does not provide a viable solution for even moderately sized networks.

An arbitrary length prefix solves problems described in detail in section 7 and are being highlighted here as well as a key part of the problem statement to be addressed. A site may not be able to delegate sufficient address space from a /64 prefix to all of its internal subnets. In this case a site may be partially operational as it is unable to number all of its subnets. An alternative would be to be able to use prefixes longer then /64 to allow multiple subnets for example /80 for numbering subnets with a mixture of hosts

that are static or DHCPv6 without worry of interoperability issues. Some operators would like the ability to have a hierarchical addressing structure and may require more than 16 bits given with a /48 allocation. In such instances longer prefix lengths would allow for additional levels of aggregation as required. It is common for some operators to have security audit requirements where they wish to know all active hosts on a /64 subnet. As /64 subnets can contain an enormous number of hosts and thus cannot be scanned as can IPv4 subnets. Operators have argued that one method to be able to scan for active hosts would be by reducing the size of the subnet. Neighbor discovery cache exhaustion when an attacker sends a large number of messages in rapid succession to hosts filling the routers ND cache is another problem with fixed length /64 size SLAAC subnets. Neighbor Discovery cache exhaustion issues are relatively common on IXP (Internet Exchange Points) where a very large number of Internet Service Providers are full mesh peering to exchange routing updates. As the number of hosts on a SLAAC subnet can be 2^64, a much smaller subnet size can drastically reduce the Neighbor Discovery cache exhaustion issues.

The goal of this document is to fix the problems related to stateless address autoconfiguration (SLAAC), current obscurities of the 64 bit prefix boundary, issues that exist today with current IPv6 addressing using manual and DHCPv6, and how variable SLAAC can now be used to fill the gaps with static and DHCPv6, and also update all standards specifications to reflect the new variable SLAAC standard making the prefix lengths variable.

## 4. Identifier and Subnet Length Statements

IPv6 router interfaces and hosts configured to use Stateful Address Autoconfiguration (SLAAC) will now support variable mask up to 128 bits consistent with static and DHCPv6. This change will allow variable SLAAC to be used on any infrastructure link from point-to-point /127 to infrastructure shared subnets from /65 to /127. All routers support routing of variable length IPv6 prefix lengths called variable length subnet masks(VLSM) up to 128 bits in length, so this variable stateless address autoconfiguration change will be in line with all interior gateway routing protocols and exterior gateway routing protocols. This change is for both Global Unicast address [RFC3587] and Unique Local Address [RFC4193]. There will be no change to the IPv6 link local address interface identifier which will remain 64 bits for link local fe80::/10 router or host interface fe80::/64 [RFC4291].

The term "Variable SLAAC" as defined in this document states that the length of the prefix can now be greater than 64 bits up to 127 bits with a corresponding shorter interface identifier. The interface

identifier will range from 64 bits to 1 bit in length. The length of the prefix can now be less than 64 bits to 3 bits in length with a corresponding longer interface identifier and can now be greater than 64 bits to 125 bits in length.

The "race to the bottom problem" - is the problem where allowing prefixes longer than 64 to be used in SLAAC will lead to 65, 66 and so on, up to 127 and 128 allocations. At that point the bottom would be reached and thus impossible to extend the network further.

One version of the "address waste" problem is: SLAAC is used in a subnet where 2^64 addresses are possible. But there are no link layers that allow as many addresses to connect on a single link. E.g. wired Ethernet allows for a few hundreds or a few thousands nodes in a switched network. Because of that, the difference up to 2^64 addresses will not be used, as such they will be wasted.

## 5. Recommendations for implementation of variable SLAAC

This document proposes a plan to provide flexibility for implementers to now have the option to use SLAAC (Stateful Address Autoconfiguration) where previously they used DHCPv6 or static. This will also open the door to interoperability and mixed device types supporting either SLAAC, static or DHCPv6 to now be able to exist on the same subnet or VLAN without risk of interoperability issues.

It is recommended to use variable length SLAAC on network infrastructure point-to-point links as well as for host subnets where historically /64 was used that now variable length SLAAC prefix can be used up to 127 bit prefix length. It is recommended that the use of variable length prefix be based on each individual IPv6 deployment requirement. If more address space is required, necessity to break up a /64 for address space management, creating an internal hierarchical addressing plan based on prefixes delegated or allocated, then variable length prefix is now an available option in the designers toolbox that now can be utilized. Changes to DHCPv6 prefix-delegation is outside of the scope of this document.

It is recommended that ISP allocations and Customer allocations per [RFC6177] and [RFC5375] not change due to this variable SLAAC proposed standard.

## 6. Recommended use cases where 64 bit prefix should be utilized

Listed below are use cases where the 64 bit prefix length MUST be adhered to and in these cases variable SLAAC feature should not be utilized.

The precise 64-bit length of the interface identifier is widely mentioned in numerous RFCs describing various aspects of IPv6. It is not straightforward to distinguish cases where this has normative impact or affects interoperability. This section aims to identify specifications that contain an explicit reference to the 64-bit length. Regardless of implementation issues, the RFCs themselves would all need to be updated if the 64-bit rule was changed, even if the updates were small, which would involve considerable time and effort.

First and foremost, the RFCs describing the architectural aspects of IPv6 addressing explicitly state, refer, and repeat this apparently immutable value: Addressing Architecture [RFC4291], IPv6 Address Assignment to End Sites [RFC6177], Reserved interface identifiers [RFC5453], and ILNP Node Identifiers [RFC6741]. Customer edge routers impose /64 for their interfaces [RFC7084]. The IPv6 Subnet Model [RFC5942] points out that the assumption of a /64 prefix length is a potential implementation error.

Numerous IPv6-over-foo documents make mandatory statements with respect to the 64-bit length of the interface identifier to be used during the Stateless Autoconfiguration. These documents include [RFC2464] (Ethernet), [RFC2467] (Fiber Distributed Data Interface (FDDI)), [RFC2470] (Token Ring), [RFC2492] (ATM), [RFC2497] (ARCnet), [RFC2590] (Frame Relay), [RFC3146] (IEEE 1394), [RFC4338] (Fibre Channel), [RFC4944] (IEEE 802.15.4), [RFC5072] (PPP), [RFC5121] [RFC5692] (IEEE 802.16), [RFC2529] (6over4), [RFC5214] (Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)), [I-D.templin-aerolink] (Asymmetric Extended Route Optimization (AERO)), [I-D.ietf-6lowpan-btle] (BLUETOOTH Low Energy), [I-D.ietf-6lo-6lobac] (IPv6 over MS/TP), and [I-D.ietf-6lo-lowpanz] (IPv6 packets over ITU-T G.9959).

To a lesser extent, the address configuration RFCs themselves may in some ways assume the 64-bit length of an interface identifier (e.g, [RFC4862] for the link-local addresses, DHCPv6 for the potentially assigned EUI- 64-based IP addresses, and Optimistic Duplicate Address Detection [RFC4429] that computes 64-bit-based collision probabilities).

The Multicast Listener Discovery Version 1 (MLDv1) [RFC2710] and MLDv2 [RFC3810] protocols mandate that all queries be sent with a link-local source address, with the exception of MLD messages sent using the unspecified address when the link-local address is tentative [RFC3590]. At the time of publication of [RFC2710], the IPv6 addressing architecture specified link-local addresses with 64-bit interface identifiers. MLDv2 explicitly specifies the use of

the fe80::/64 link-local prefix and bases the querier election algorithm on the link-local subnet prefix of length /64.

The "IPv6 Flow Label Specification" [RFC6437] gives an example of a 20-bit hash function generation, which relies on splitting an IPv6 address in two equally sized, 64-bit-length parts.

The basic transition mechanisms [RFC4213] refer to interface identifiers of length 64 for link-local addresses; other transition mechanisms such as Teredo [RFC4380] assume the use of interface identifiers of length 64. Similar assumptions are found in 6to4 [RFC3056] and 6rd [RFC5969]. Translation-based transition mechanisms such as NAT64 and NPTv6 have some dependency on prefix length, discussed below.

The proposed method [RFC7278] of extending an assigned /64 prefix from a smartphone's cellular interface to its WiFi link relies on prefix length, and implicitly on the length of the interface identifier, to be valued at 64.

The Cryptographically Generated Addresses (CGA) and Hash-Based Addresses (HBA) specifications rely on the 64-bit identifier length (see below), as do the Privacy extensions [RFC4941] and some examples in "Internet Key Exchange Version 2 (IKEv2)" [RFC7296].

464XLAT [RFC6877] explicitly mentions acquiring /64 prefixes. However, it also discusses the possibility of using the interface address on the device as the end point for the traffic, thus potentially removing this dependency.

[RFC2526] reserves a number of subnet anycast addresses by reserving some anycast interface identifiers. An anycast interface identifier so reserved cannot be less than 7 bits long. This means that a subnet prefix length longer than /121 is not possible, and a subnet of exactly /121 would be useless since all its identifiers are reserved. It also means that half of a /120 is reserved for anycast. This could of course be fixed in the way described for /127 in [RFC6164], i.e., avoiding the use of anycast within a /120 subnet. Note that support for "on-link anycast" is a standard IPv6 neighbor discovery capability [RFC4861] [RFC7094]; therefore, applications and their developers would expect it to be available.

The Mobile IP home network models [RFC4887] rely heavily on the /64 subnet length and assume a 64-bit interface identifier.

- Multicast: [RFC3306] defines a method for generating IPv6 multicast group addresses based on unicast prefixes. This method assumes a longest prefix of 64 bits. If a longer prefix is used,

there is no way to generate a specific multicast group address using this method. In such cases, the administrator would need to use an "artificial" prefix from within their allocation (a /64 or shorter) from which to generate the group address. This prefix would not correspond to a real subnet.

- Similarly, [RFC3956], which specifies the Embedded Rendezvous Point (RP)) allowing IPv6 multicast rendezvous point addresses to be embedded in the multicast group address, would also fail, as the scheme assumes a maximum prefix length of 64 bits.

- CGA: The Cryptographically Generated Address format [RFC3972] is heavily based on a /64 interface identifier. [RFC3972] has defined a detailed algorithm showing how to generate a 64-bit interface identifier from a public key and a 64-bit subnet prefix. Changing the /64 boundary would certainly invalidate the current CGA definition. However, the CGA might benefit in a redefined version if more bits are used for interface identifiers (which means shorter prefix length). For now, 59 bits are used for cryptographic purposes. The more bits are available, the stronger CGA could be. Conversely, longer prefixes would weaken CGA.

- NAT64: Both stateless NAT64 [RFC6052] and stateful NAT64 [RFC6146] are flexible for the prefix length. [RFC6052] has defined multiple address formats for NAT64. In Section 2 of "IPv4-Embedded IPv6 Address Prefix and Format" [RFC6052], the network-specific prefix could be one of /32, /40, /48, /56, /64, and /96. The remaining part of the IPv6 address is constructed by a 32-bit IPv4 address, an 8-bit u byte and a variable length suffix (there is no u byte and suffix in the case of the 96-bit Well-Known Prefix). NAT64 is therefore OK with a subnet boundary out to /96 but not longer.

- NPTv6: IPv6-to-IPv6 Network Prefix Translation [RFC6296] is also bound to /64 boundary. NPTv6 maps a /64 prefix to another /64 prefix. When the NPTv6 Translator is configured with a /48 or shorter prefix, the 64-bit interface identifier is kept unmodified during translation. However, the /64 boundary might be changed as long as the "inside" and "outside" prefixes have the same length.

- ILNP: Identifier-Locator Network Protocol (ILNP) [RFC6741] is designed around the /64 boundary, since it relies on locally unique 64-bit node identifiers (in the interface identifier field). While a redesign to use longer prefixes is not inconceivable, this would need major changes to the existing specification for the IPv6 version of ILNP.

- Shim6: The Multihoming Shim Protocol for IPv6 (Shim6) [RFC5533] in its insecure form treats IPv6 addresses as opaque 128-bit objects. However, to secure the protocol against spoofing, it is essential to either use CGAs (see above) or HBAs [RFC5535]. Like CGAs, HBAs are generated using a procedure that assumes a 64-bit identifier. Therefore, in effect, secure shim6 is affected by the /64 boundary exactly like CGAs.

- Duplicate address risk: If SLAAC was modified to work with shorter interface identifiers, the statistical risk of hosts choosing the same pseudo- random identifier [RFC7217] would increase correspondingly. The practical impact of this would range from slight to dramatic, depending on how much the interface identifier length was reduced. In particular, a /120 prefix would imply an 8-bit interface identifier and address collisions would be highly probable.

- The link-local prefix: While [RFC4862] is careful not to define any specific length of link-local prefix within fe80::/10, the addressing architecture [RFC4291] does define the link-local interface identifier length to be 64 bits. If different hosts on a link used interface identifiers of different lengths to form a link-local address, there is potential for confusion and unpredictable results. Typically today the choice of 64 bits for the link-local interface identifier length is hard-coded per interface, in accordance with the relevant IPv6-over-foo specification, and systems behave as if the link-local prefix was actually fe80::/64. There might be no way to change this except conceivably by manual configuration, which will be impossible if the host concerned has no local user interface.

## 7. Reasons for longer than 64 bit prefix length

In this section we are providing the justification for longer prefixes and shorter interface identifiers essentially variable SLAAC.

### 7.1. Insufficient Address Space Delegated

A site may not be delegated a sufficiently generous prefix from which to allocate a /64 prefix to all of its internal subnets. In this case, the site may either determine that it does not have enough address space to number all its network elements and thus, at the very best, be only partially operational, or it may choose to use internal prefixes longer than /64 to allow multiple subnets and the hosts within them to be configured with addresses.

In this case, the site might choose, for example, to use a /80 per subnet in combination with hosts using either manually configured addressing or DHCPv6 [RFC3315].

Scenarios that have been suggested where an insufficient prefix might be delegated include home or small office networks, vehicles, building services, and transportation services (e.g., road signs). It should be noted that the homenet architecture text [RFC7368] states that Customer Premises Equipment (CPE) should consider the lack of sufficient address space to be an error condition, rather than using prefixes longer than /64 internally.

Another scenario occasionally suggested is one where the Internet address registries actually begin to run out of IPv6 prefix space, such that operators can no longer assign reasonable prefixes to users in accordance with [RFC6177]. It is sometimes suggested that assigning a prefix such as /48 or /56 to every user site (including the smallest) as recommended by [RFC6177] is wasteful. In fact, the currently released unicast address space, 2000::/3, contains 35 trillion /48 prefixes ((2**45 = 35,184,372,088,832), of which only a small fraction have been allocated. Allowing for a conservative estimate of allocation efficiency, i.e., an HD-ratio of 0.94 [RFC4692], approximately 5 trillion /48 prefixes can be allocated. Even with a relaxed HD-ratio of 0.89, approximately one trillion /48 prefixes can be allocated. Furthermore, with only 2000::/3 currently committed for unicast addressing, we still have approximately 85% of the address space in reserve. Thus, there is no objective risk of prefix depletion by assigning /48 or /56 prefixes even to the smallest sites.

### 7.2. Hierarchical Addressing

Some operators have argued that more prefix bits are needed to allow an aggregated hierarchical addressing scheme within a campus or corporate network. However, if a campus or enterprise gets a /48 prefix (or shorter), then that already provides 16 bits for hierarchical allocation. In any case, flat IGP routing is widely and successfully used within rather large networks, with hundreds of routers and thousands of end systems. Therefore, there is no objective need for additional prefix bits to support hierarchy and aggregation within enterprises.

### 7.3. Audit Requirement

Some network operators wish to know and audit nodes that are active on a network, especially those that are allowed to communicate off-link or off-site. They may also wish to limit the total number of active addresses and sessions that can be sourced from a particular

host, LAN, or site, in order to prevent potential resource-depletion attacks or other problems spreading beyond a certain scope of control. It has been argued that this type of control would be easier if only long network prefixes with relatively small numbers of possible hosts per network were used, reducing the discovery problem. However, such sites most typically operate using DHCPv6, which means that all legitimate hosts are automatically known to the DHCPv6 servers, which is sufficient for audit purposes. Such hosts could, if desired, be limited to a small range of interface identifier values without changing the /64 subnet length. Any hosts inadvertently obtaining addresses via SLAAC can be audited through Neighbor Discovery (ND) logs.

### 7.4. Concerns over ND Cache Exhaustion

A site may be concerned that it is open to ND cache exhaustion attacks [RFC3756], whereby an attacker sends a large number of messages in rapid succession to a series of (most likely inactive) host addresses within a specific subnet. Such an attack attempts to fill a router's ND cache with ND requests pending completion, which results in denying correct operation to active devices on the network.

One potential way to mitigate this attack would be to consider using a /120 prefix, thus limiting the number of addresses in the subnet to be similar to an IPv4 /24 prefix, which should not cause any concerns for ND cache exhaustion. Note that the prefix does need to be quite long for this scenario to be valid. The number of theoretically possible ND cache slots on the segment needs to be of the same order of magnitude as the actual number of hosts. Thus, small increases from the /64 prefix length do not have a noticeable impact; even 2^32 potential entries, a factor of two billion decrease compared to 2^64, is still more than enough to exhaust the memory on current routers. Given that most link-layer mappings cause SLAAC to assume a 64-bit network boundary, in such an approach hosts would likely need to use DHCPv6 or be manually configured with addresses.

It should be noted that several other mitigations of the ND cache attack are described in [RFC6583], and that limiting the size of the cache and the number of incomplete entries allowed would also defeat the attack. For the specific case of a point-to-point link between routers, this attack is indeed mitigated by a /127 prefix [RFC6164].

### 7.5. Longer prefixes lengths used for embedding information

Ability to utilize the longer than 64 bit prefixes to be able to embed geographic or other information into the prefix that could be

valuable to the IPv6 addressing architecture providing more flexibility to the operator.

## 8. Greater than 64 bit prefix usage by ISPs is strictly prohibited

The RA flag S bit setting proposed by this draft for greater or less then /64 bit prefix feature is strictly for enterprises and broadband subscriber customer use only. In the broadband space this feature is to be used only by home broadband users or Small Office Home Office broadband users. ISPs can use DHCPv6-PD to send greater than /64 prefix advertisement message to the CE router. However, ISPs MUST NOT set the RA flag S bit, and send greater than /64 prefix down to the host. With this draft, no changes will be made to any of the current IPv6 prefix allocation guidelines for customers prefixes sizes sent by ISPs. In the enterprise space any router is allowed to set the RA flag S bit for greater then /64 prefix allocation. This provides tremendous flexibility for enterprises as well as broadband subscriber customers to segment and further extend a single /64 allocation. Enterprise use case allows for added further segmentation granularity and functionality for endpoints devices. Broadband subscriber use cases with the proliferation of IoT and 6lo devices in the home or small office use case, allows for added further segmentation granularity and functionality for endpoints devices. This change of stateful address auto configuration to now allow for greater then 64 bit prefix length is not being done because there is wasted space with a /64 prefix length or that there is any even remote possibility of running out of address space. The reason for restriction of use of this feature by ISP's is also because it puts the customer at the shorter end of the stick with smaller allocation, which you could be interpreted as biased or unfair to smaller customers that cannot afford the larger space as large companies. This would be another form of inequality between customers and service provider similar to net neutrality discussions and fairness in quality of service. ISPs have essentially nothing to gain by advertising smaller prefixes with greater than /64 prefixes allocations as address space is in abundance. On the other hand for ISPs' OPEX costs could be increased and so much less cost effective to administer greater then /64 variable length prefixes versed fixed /64 size prefix would be much more challenging and cost prohibitive to manage.

## 9. Comparison of Static, SLAAC, DHCPv6 and Variable SLAAC

- Static - IPv6 address and Default Gateway:

  Pros:

  + Deactivation of RA processing

+ Good resistance against RA attack

Cons:

+ Operational impact in configuring interface manually

+ Network dynamics might require renumbering which needs work

o Static - IPv6 address and Default Route via RA

Pros:

+ Does not require disabling RA processing

+ Works better with FHRP router redundancy

Cons:

+ RA related security issues combat with RA Guard

o DHCPv6 [RFC3315]

Pros:

+ Centralized provisioning of IPv6 addressing

+ IPv6, DNS, NTP can all be distributed

Cons:

+ Administrative overhead of managing DHCPv6 server

+ Caveats with redundancy and split scopes required for failover. Split scope and failover is resolved with DHCPv6 Failover protocol [RFC8156]

+ RA related security issues combat with RA Guard

o SLAAC [RFC7217] Stable Random station-id with privacy and [RFC8064] Recommendations for Stable interfae identifier

Pros:

+ Automatic provisioning IPv6 address to hosts

+ [RFC7217] Stable Random station-id with privacy extensions

Cons:

+ RA related security issues combat with RA Guard

o Variable SLAAC with [RFC7217] Stable Random station-id with privacy and [RFC8064] Recommendations for Stable interfae identifier

Pros:

+ Automatic provisioning IPv6 address to hosts

+ [RFC7217] Stable Random station-id with privacy extensions

Cons:

+ RA related security issues combat with RA Guard

+ Security is reduced with longer prefixes and shorter stable random station-id

IPv6 address deployment summary statement.

DHCPv6 [RFC3315] state machine introduces a large number of messaging packets with Normal mode, four messages called solicit, advertise, request and reply. DHCPv6 Rapid Commit mode reduces the messages from four to two messages only solicit and reply. DHCPv6 Normal mode is the Default. It is recommended to use DHCPv6 Rapid mode [RFC4039] in "high mobility" networks where clients come and go often. The overhead of four messages might not be required so two messages might enough to accommodate. However, if you have multiple DHCPv6 servers for redundancy then you need to use DHCPv6 Normal mode. If you have subnets where there are a large flat user subnets with a very large number of hosts and redundancy is required and DHCPv6 Normal mode is utilized, DHCPv6 messaging is exacerbated exponentially as the subnets flatten out further and further. As the paradigm shifts and IPv4 is eliminated as hosts subnets change to "IPv6-ONLY" subnets, the coupling of IPv4 with IPv6 Dual stack dependency is eliminated, thus removing the shackles pinning IPv6 to smaller many IPv4 subnets. This change allows IPv6 subnets to become very large and flat with the only limiting factor being the L2 switch infrastructure. In many cases Dual stacked implementations with 100's of subnets may change to a single "IPV6 ONLY" subnet. As "IPV6-ONLY" subnets will soon become the future direction of all user access infrastructure, we need a viable solution that will accommodate these very large flat subnets. The problem with DHCPv6 is that once the "M" managed bit is set to "1", all hosts on the subnet cache the Managed IP "M bit" and changes host to DHCPv6 stateful mode. Higher probability of rouge devices such as printers or other appliances misbehaving with IPv6 enabled by default, now in DHCPv6 mode, spewing of millions of DHCPv6

messages that can now impact the router control plane processing of packets. This can be alleviated with special custom Control plane policer policy, however now adds complexity and administrative overhead to DHCPv6 deployments. Enterprises and Service Providers require a viable IPv6 deployment solution that can accommodate the shortfalls of both static and DHCPv6 addressing. Static addressing due to administrative overhead of manual assignment does not provide a viable solution for even moderately sized networks. Variable SLAAC now has the ability to fill the gaps outlined with DHCPv6 and static that can now be used as a viable ubiquitous all encompassing solution for IPv6 address deployments.

## 10. Variable SLAAC Use Cases

This section describes real world use cases of variable slaac that cannot be done today and with fixed 64 bit prefix lengths.

### 10.1. Permission-less Extension of the Network

Permission-less extensions of the network with new links (and by implication with new routers) are not supported.

The lack of possibility to realize a permission-less extension of the network is an important problem. The problem appears at the edge of the network. The permission is 'granted' for end users situated at the edge of the network. This permission is 'granted' by advertising a prefix of length 64, typically. This prefix is set in the PIO option in an RA. The end user receives this prefix, forms an address, and is able to connect to the Internet. However, the end user has no permission to further extend the network. Although s/he is able to form subsequent prefixes of a length of, say 65, and further advertise it down in the extension of the network, no other Host in that extension of the network is able to use that advertisement; a Host can not form an address with a prefix length 65 by using SLAAC. The linux error text reported in the kernel log upon reception of a plen 65 is "illegal" (or similar).

### 10.2. Private Networks

Private networks such as Service Provider core not accessable by customers and enterprises where all hosts are trusted are the primary use case for variable SLAAC as the shorter interface identifier does not create any security issues with not having a longer 64 bit interface identifier for privacy extensions stable interface identifier [RFC8084] due to all hosts being inherently trusted. Private internal networks such as corporate intranets traditionally have always used static IPv6 addressing for infrastructure. This manual IPv6 address assignment process for network infrastructure

links can take long lead times to complete deployment. By changing the behavior of SLAAC to support variable length prefix and interface identifier allows SLAAC to be used programmatically to deploy to large scale IPv6 networks with thousands of point-to-point links. Note that network infrastructure technically does not require IPv6 addressing due to IPv6 next hop being a link local address for IGP routing protocols such as OSPF and ISIS as well as the link local address can be the peer IPv6 address for exterior gateway routing protocols such as BGP. However for hop by hop ping and traceroute capability to have IPv6 reachability at each hop for troubleshooting jitter, latency and drops it is an IPv6 recommended best practice to configure IPv6 address on all infrastructure interfaces.

### 10.3. Mobile IPv6

Old MIP6 (Mobile IPv6) Working Group and old Nemo Working Group's routing solution scenarios related to Mobile IPv6 ([RFC3775]) (note: nowadays most MIP-related activity is in DMM WG) where the mobile endpoint can now obtain from the home agent variable SLAAC address and not 64 bit prefix /64 address. This maybe useful in cases where a /64 can now be managed from an addressing perspective and subdivided into blocks for manageability of MIP6 endpoints instead of allocating a single /64 per endpoint.

### 10.4. Home and SOHO

Home and SOHO (Small Office and Home Office) environments where internet access uses a broadband service provider single or dual homed scenario. In those such Home networking Homenet environments where HNCP (Home Network Control Protocol [RFC7788] SADR (Source Address Dependent Routing) are deployed for automatic configuration for LAN WIFI endpoint subnets can also now take advantage of variable length SLAAC in deployment scenarios. In cases where multiple routers are deployed in a home environment where routing prefix reachability needs to be advertised where Bable [RFC6126] routing protocol is utilized in those cases variable SLAAC can also be utilized to break up a /64 into multiple smaller subnets.

### 10.5. 3GPP V2I and V2V networking

In V2I networking (with 3GPP or with IEEE 802.11bd) the vehicle receives a /64 prefix from the cellular network (or from a Road-Side Unit). This /64 prefix can be used to form one address for the egress interface of the Mobile Router (IP On-Board Unit), but can not be used to form IP addresses for other hosts in the vehicle.

3GPP V2V networking use cases where a /56 is allocated to the 4G modem and a /64 is delegated to downstream devices within the

automobile now the /64 prefixes can be sub divided into smaller prefix lengths of /65-/128. This provides additional granularity to use cases.

### 10.6. 6lo

6lo Working IPv6 over Network Constrained nodes working group use cases. Use cases for IoT devices where have limited network access requirements could now take advantage of variable SLAAC longer prefixes lengths /65-/128.

### 10.7. Large ISP's backbone POP

Large ISP backbone POPs such as IXPs where many carriers share the same backbone and ND cache exhaustion may occur due to /64 subnet size. One mitigation technique employed is the use of an ARP Sponge for IPv4 or Layer 2 multicast rate limiters for IPv6. In those particular cases a longer prefix static or variable SLAAC subnet could be utilized to reduce the maximum number of hosts on the subnet.

## 11. Variable SLAAC implementation using RA Flag

[RFC5175] currently defines the flags in the NDP Router Advertisement message and these flags are registered in the IANA IPv6 ND Router Advertisement flags Registry [IANA-RF]. This currently contains the following one-bit flags defined in published RFCs:

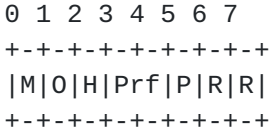


Figure 2: Format of flags in Router Advertisement message

- M: Managed Address Configuration Flag [RFC4861]
- O: Other Configuration Flag [RFC4861]
- H: Mobile IPv6 Home Agent Flag [RFC3775]
- Prf: Router Selection Preferences [RFC4191]
- P: Neighbor Discovery Proxy Flag [RFC4389]
- R: Reserved

This document defines bit 6 to be the Variable SLAAC Flag:

o S: SLAAC Variable length interface identifier Flag

This flag has two values. These are:

o 0: Variable length interface identifier is disabled

o 1: Variable length interface identifier is enabled (ignored by hosts not supporting)

[RFC5175] requires that unused flag bits be set to zero. Therefore, a router that does not support the new flag will not appear to assert that the PIO list prefix list advertised does not support variable length interface identifier.

Hosts receiving the Router Advertisement SHOULD only process this flag if the advertising router is a Default Router. Specifically, if the Lifetime field in the Router Advertisement is not zero, otherwise it SHOULD be ignored.

Note that although this mechanism uses one of only two reserved flag bits in the RA, an extension mechanism is defined in Section 4 of [RFC5175] in case additional flags are ever required for future extensions. It should be noted that since [RFC5175] was published in 2008, no new RA flags have been assigned in the IANA registry.

This draft precludes the use of EUI64 based, 64 bit fixed length interface identifier generation algorithms, and allows the use of any standard variable interface identifier generation algorithm for the auto generating variable length interface identifier less than 64 bits for example [RFC4941] Privacy Extensions for Stateless Address Autoconfiguration in IPv6 or [RFC7217] Semantically opaque interface identifier with SLAAC privacy extension algorithm for stable variable length interface identifier per [RFC8064]. In this particular case the prefix will be greater than 64 bits and the stable interface identifier will be less than 64 bits.

This draft precludes the use of EUI64 based, 64 bit fixed length interface identifier generation algorithms, and allows the use of any standard variable interface identifier generation algorithm for the auto generating variable length interface identifier greater than 64 bits for example [RFC4941] Privacy Extensions for Stateless Address Autoconfiguration in IPv6 or [RFC7217] Semantically opaque interface identifier with SLAAC privacy extension algorithm for stable variable length interface identifier per [RFC8064]. In this particular case the prefix will be less than 64 bits and the stable interface identifier will be less than 64 bits.

Draft rfc4941bis Privacy Extension for SLAAC using [RFC4086] Pseudo-Random Number Generator(PRNG) can also be used as a possible method of generating greater then 64 bit or less then 64 bit interface identifier automatically since stated in the draft that the interface identifier can be generated for any arbitrary length. https://datatracker.ietf.org/doc/draft-ietf-6man-rfc4941bis/

## 12. Applicability Statements

The RA flag option for variable length interface identifier is designed to allow administrators send variable length prefixes in the PIO list advertisement to the host and hosts supporting this variable interface identifier option would be able to process the flag and use the prefix with variable interface identifier in the PIO list .

Administrators MUST only use this variable length interface identifier flag configured on the router to signal processing of the longer prefix if they have longer then 64 bit prefix configured on the router.

## 13. Router and Operational Considerations

Default IPv6 routers that have the variable interface identifier with longer than 64 bit prefixes SHOULD be configured by the administrator to set the variable SLAAC flag to 1. In all other cases the flag SHOULD NOT be set to 1.

The intent is that the administrator of the router configures the router to set the variable SLAAC flag if and only if she/he wants to tell the hosts on the link that the prefixes being sent are greater then 64 bits and shorter then the standard 64 bit interface identifier. This is a configuration flag, it is not something that the router decides on its own. Routers MAY log a configuration error if the flag is set and the prefix is not longer the 64 bits and the interface identifier is not shorter then 64 bits.

Routers implementing this document SHOULD log to system or network management inconsistent setting of the variable SLAAC flag. This extends the behavior specified in Section 6.2.7 of [RFC4861].

## 14. Host Behavior Considerations

Host operating system support will be backwards compatible so host that do not support the flag will ignore the variable SLAAC flag being set to 1. In that scenario the router supports the variable length prefix option and would be configured with the flag set and would send the variable length prefix to the host, however the host not supporting the flag will not accept the prefix as it is not 64

bit length and as it is unable to process the flag. Hosts that support the variable SLAAC RA flag MUST have a configuration option to ignore or process the flag. The motivation for this configuration option is for hosts that are capable of processing the variable SLAAC flag to only act on the flag if they are configured to do so.

If there are multiple IPv6 default routers on a link, they might send different values of the flag. If at least one IPv6 default router sends the flag with value 1, the host supporting the flag will receive will receive and process the flag and accept the PIO list with variable prefixes. If all IPv6 default routers send the flag with value 1 the host will receive and process the prefix and flag from all routers sending the RA.

## 15. Security Considerations

The administrator should be aware to maintain 64 bit interface identifier for privacy when connected directly to the internet so that entropy for optimal heuristics are maintained for security.

Variable length interface identifier shorter then 64 bits should be only used within corporate intranets and private networks where all hosts are trusted.

In all cases where the host is on a public network for privacy concerns to avoid traceability variable interface identifier MUST never be utilized.

## 16. IANA Considerations

IANA is requested to assign the new Router Advertisement flag defined in Section 5 of this document. Bit 6 is the next available bit in this registry, IANA is requested to use this bit unless there is a reason to use another bit in this registry.

IANA is also requested to register this new flag bit in the IANA IPv6 ND Router Advertisement flags Registry [IANA-RF].

## 17. Contributors

Contributors.

## 18. Acknowledgements

## 19. References

### 19.1. Normative References

[I-D.bourbaki-6man-classless-ipv6]
Bourbaki, N., "IPv6 is Classless", draft-bourbaki-6man-classless-ipv6-05 (work in progress), April 2019.

[I-D.ietf-6lo-6lobac]
Lynn, K., Martocci, J., Neilson, C., and S. Donaldson, "Transmission of IPv6 over MS/TP Networks", draft-ietf-6lo-6lobac-08 (work in progress), March 2017.

[I-D.ietf-6lo-lowpanz]
Brandt, A. and J. Buron, "Transmission of IPv6 packets over ITU-T G.9959 Networks", draft-ietf-6lo-lowpanz-08 (work in progress), October 2014.

[I-D.ietf-6lowpan-btle]
Nieminen, J., Savolainen, T., Isomaki, M., Patil, B., Shelby, Z., and C. Gomez, "Transmission of IPv6 Packets over BLUETOOTH Low Energy", draft-ietf-6lowpan-btle-12 (work in progress), February 2013.

[I-D.mishra-v6ops-variable-slaac-problem-stmt]
Mishra, G., Petrescu, A., Kottapalli, N., Mudric, D., and D. Shytyi, "SLAAC with prefixes of arbitrary length in PIO (Variable SLAAC)", draft-mishra-v6ops-variable-slaac-problem-stmt-00 (work in progress), October 2020.

[I-D.templin-aerolink]
Templin, F., "Asymmetric Extended Route Optimization (AERO)", draft-templin-aerolink-82 (work in progress), May 2018.

[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>.

[RFC2450] Hinden, R., "Proposed TLA and NLA Assignment Rule", RFC 2450, DOI 10.17487/RFC2450, December 1998, <https://www.rfc-editor.org/info/rfc2450>.

[RFC2464] Crawford, M., "Transmission of IPv6 Packets over Ethernet Networks", RFC 2464, DOI 10.17487/RFC2464, December 1998, <https://www.rfc-editor.org/info/rfc2464>.

[RFC2467] Crawford, M., "Transmission of IPv6 Packets over FDDI Networks", RFC 2467, DOI 10.17487/RFC2467, December 1998, <https://www.rfc-editor.org/info/rfc2467>.

[RFC2470] Crawford, M., Narten, T., and S. Thomas, "Transmission of IPv6 Packets over Token Ring Networks", RFC 2470, DOI 10.17487/RFC2470, December 1998, <https://www.rfc-editor.org/info/rfc2470>.

[RFC2492] Armitage, G., Schulter, P., and M. Jork, "IPv6 over ATM Networks", RFC 2492, DOI 10.17487/RFC2492, January 1999, <https://www.rfc-editor.org/info/rfc2492>.

[RFC2497] Souvatzis, I., "Transmission of IPv6 Packets over ARCnet Networks", RFC 2497, DOI 10.17487/RFC2497, January 1999, <https://www.rfc-editor.org/info/rfc2497>.

[RFC2526] Johnson, D. and S. Deering, "Reserved IPv6 Subnet Anycast Addresses", RFC 2526, DOI 10.17487/RFC2526, March 1999, <https://www.rfc-editor.org/info/rfc2526>.

[RFC2529] Carpenter, B. and C. Jung, "Transmission of IPv6 over IPv4 Domains without Explicit Tunnels", RFC 2529, DOI 10.17487/RFC2529, March 1999, <https://www.rfc-editor.org/info/rfc2529>.

[RFC2590] Conta, A., Malis, A., and M. Mueller, "Transmission of IPv6 Packets over Frame Relay Networks Specification", RFC 2590, DOI 10.17487/RFC2590, May 1999, <https://www.rfc-editor.org/info/rfc2590>.

[RFC2710] Deering, S., Fenner, W., and B. Haberman, "Multicast Listener Discovery (MLD) for IPv6", RFC 2710, DOI 10.17487/RFC2710, October 1999, <https://www.rfc-editor.org/info/rfc2710>.

[RFC3056] Carpenter, B. and K. Moore, "Connection of IPv6 Domains via IPv4 Clouds", RFC 3056, DOI 10.17487/RFC3056, February 2001, <https://www.rfc-editor.org/info/rfc3056>.

[RFC3146] Fujisawa, K. and A. Onoe, "Transmission of IPv6 Packets over IEEE 1394 Networks", RFC 3146, DOI 10.17487/RFC3146, October 2001, <https://www.rfc-editor.org/info/rfc3146>.

[RFC3177] IAB and IESG, "IAB/IESG Recommendations on IPv6 Address Allocations to Sites", RFC 3177, DOI 10.17487/RFC3177, September 2001, <https://www.rfc-editor.org/info/rfc3177>.

[RFC3306] Haberman, B. and D. Thaler, "Unicast-Prefix-based IPv6 Multicast Addresses", RFC 3306, DOI 10.17487/RFC3306, August 2002, <https://www.rfc-editor.org/info/rfc3306>.

[RFC3315] Droms, R., Ed., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, DOI 10.17487/RFC3315, July 2003, <https://www.rfc-editor.org/info/rfc3315>.

[RFC3513] Hinden, R. and S. Deering, "Internet Protocol Version 6 (IPv6) Addressing Architecture", RFC 3513, DOI 10.17487/RFC3513, April 2003, <https://www.rfc-editor.org/info/rfc3513>.

[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>.

[RFC3590] Haberman, B., "Source Address Selection for the Multicast Listener Discovery (MLD) Protocol", RFC 3590, DOI 10.17487/RFC3590, September 2003, <https://www.rfc-editor.org/info/rfc3590>.

[RFC3627] Savola, P., "Use of /127 Prefix Length Between Routers Considered Harmful", RFC 3627, DOI 10.17487/RFC3627, September 2003, <https://www.rfc-editor.org/info/rfc3627>.

[RFC3756] Nikander, P., Ed., Kempf, J., and E. Nordmark, "IPv6 Neighbor Discovery (ND) Trust Models and Threats", RFC 3756, DOI 10.17487/RFC3756, May 2004, <https://www.rfc-editor.org/info/rfc3756>.

[RFC3775] Johnson, D., Perkins, C., and J. Arkko, "Mobility Support in IPv6", RFC 3775, DOI 10.17487/RFC3775, June 2004, <https://www.rfc-editor.org/info/rfc3775>.

[RFC3810] Vida, R., Ed. and L. Costa, Ed., "Multicast Listener Discovery Version 2 (MLDv2) for IPv6", RFC 3810, DOI 10.17487/RFC3810, June 2004, <https://www.rfc-editor.org/info/rfc3810>.

[RFC3956] Savola, P. and B. Haberman, "Embedding the Rendezvous Point (RP) Address in an IPv6 Multicast Address", RFC 3956, DOI 10.17487/RFC3956, November 2004, <https://www.rfc-editor.org/info/rfc3956>.

[RFC3972] Aura, T., "Cryptographically Generated Addresses (CGA)", RFC 3972, DOI 10.17487/RFC3972, March 2005, <https://www.rfc-editor.org/info/rfc3972>.

[RFC4039] Park, S., Kim, P., and B. Volz, "Rapid Commit Option for the Dynamic Host Configuration Protocol version 4 (DHCPv4)", RFC 4039, DOI 10.17487/RFC4039, March 2005, <https://www.rfc-editor.org/info/rfc4039>.

[RFC4086] Eastlake 3rd, D., Schiller, J., and S. Crocker, "Randomness Requirements for Security", BCP 106, RFC 4086, DOI 10.17487/RFC4086, June 2005, <https://www.rfc-editor.org/info/rfc4086>.

[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>.

[RFC4213] Nordmark, E. and R. Gilligan, "Basic Transition Mechanisms for IPv6 Hosts and Routers", RFC 4213, DOI 10.17487/RFC4213, October 2005, <https://www.rfc-editor.org/info/rfc4213>.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, DOI 10.17487/RFC4291, February 2006, <https://www.rfc-editor.org/info/rfc4291>.

[RFC4338] DeSanti, C., Carlson, C., and R. Nixon, "Transmission of IPv6, IPv4, and Address Resolution Protocol (ARP) Packets over Fibre Channel", RFC 4338, DOI 10.17487/RFC4338, January 2006, <https://www.rfc-editor.org/info/rfc4338>.

[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>.

[RFC4389] Thaler, D., Talwar, M., and C. Patel, "Neighbor Discovery Proxies (ND Proxy)", RFC 4389, DOI 10.17487/RFC4389, April 2006, <https://www.rfc-editor.org/info/rfc4389>.

[RFC4429] Moore, N., "Optimistic Duplicate Address Detection (DAD) for IPv6", RFC 4429, DOI 10.17487/RFC4429, April 2006, <https://www.rfc-editor.org/info/rfc4429>.

[RFC4548] Gray, E., Rutemiller, J., and G. Swallow, "Internet Code Point (ICP) Assignments for NSAP Addresses", RFC 4548, DOI 10.17487/RFC4548, May 2006, <https://www.rfc-editor.org/info/rfc4548>.

[RFC4692] Huston, G., "Considerations on the IPv6 Host Density Metric", RFC 4692, DOI 10.17487/RFC4692, October 2006, <https://www.rfc-editor.org/info/rfc4692>.

[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>.

[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>.

[RFC4887] Thubert, P., Wakikawa, R., and V. Devarapalli, "Network Mobility Home Network Models", RFC 4887, DOI 10.17487/RFC4887, July 2007, <https://www.rfc-editor.org/info/rfc4887>.

[RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 4941, DOI 10.17487/RFC4941, September 2007, <https://www.rfc-editor.org/info/rfc4941>.

[RFC4944] Montenegro, G., Kushalnagar, N., Hui, J., and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks", RFC 4944, DOI 10.17487/RFC4944, September 2007, <https://www.rfc-editor.org/info/rfc4944>.

[RFC5072] Varada, S., Ed., Haskins, D., and E. Allen, "IP Version 6 over PPP", RFC 5072, DOI 10.17487/RFC5072, September 2007, <https://www.rfc-editor.org/info/rfc5072>.

[RFC5121] Patil, B., Xia, F., Sarikaya, B., Choi, JH., and S. Madanapalli, "Transmission of IPv6 via the IPv6 Convergence Sublayer over IEEE 802.16 Networks", RFC 5121, DOI 10.17487/RFC5121, February 2008, <https://www.rfc-editor.org/info/rfc5121>.

[RFC5175] Haberman, B., Ed. and R. Hinden, "IPv6 Router Advertisement Flags Option", RFC 5175, DOI 10.17487/RFC5175, March 2008, <https://www.rfc-editor.org/info/rfc5175>.

[RFC5214] Templin, F., Gleeson, T., and D. Thaler, "Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)", RFC 5214, DOI 10.17487/RFC5214, March 2008, <https://www.rfc-editor.org/info/rfc5214>.

[RFC5375] Van de Velde, G., Popoviciu, C., Chown, T., Bonness, O., and C. Hahn, "IPv6 Unicast Address Assignment Considerations", RFC 5375, DOI 10.17487/RFC5375, December 2008, <https://www.rfc-editor.org/info/rfc5375>.

[RFC5453] Krishnan, S., "Reserved IPv6 Interface Identifiers", RFC 5453, DOI 10.17487/RFC5453, February 2009, <https://www.rfc-editor.org/info/rfc5453>.

[RFC5533] Nordmark, E. and M. Bagnulo, "Shim6: Level 3 Multihoming Shim Protocol for IPv6", RFC 5533, DOI 10.17487/RFC5533, June 2009, <https://www.rfc-editor.org/info/rfc5533>.

[RFC5535] Bagnulo, M., "Hash-Based Addresses (HBA)", RFC 5535, DOI 10.17487/RFC5535, June 2009, <https://www.rfc-editor.org/info/rfc5535>.

[RFC5692] Jeon, H., Jeong, S., and M. Riegel, "Transmission of IP over Ethernet over IEEE 802.16 Networks", RFC 5692, DOI 10.17487/RFC5692, October 2009, <https://www.rfc-editor.org/info/rfc5692>.

[RFC5942] Singh, H., Beebee, W., and E. Nordmark, "IPv6 Subnet Model: The Relationship between Links and Subnet Prefixes", RFC 5942, DOI 10.17487/RFC5942, July 2010, <https://www.rfc-editor.org/info/rfc5942>.

[RFC5969] Townsley, W. and O. Troan, "IPv6 Rapid Deployment on IPv4 Infrastructures (6rd) -- Protocol Specification", RFC 5969, DOI 10.17487/RFC5969, August 2010, <https://www.rfc-editor.org/info/rfc5969>.

[RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, DOI 10.17487/RFC6052, October 2010, <https://www.rfc-editor.org/info/rfc6052>.

[RFC6126] Chroboczek, J., "The Babel Routing Protocol", RFC 6126, DOI 10.17487/RFC6126, April 2011, <https://www.rfc-editor.org/info/rfc6126>.

[RFC6146] Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC 6146, DOI 10.17487/RFC6146, April 2011, <https://www.rfc-editor.org/info/rfc6146>.

[RFC6164] Kohno, M., Nitzan, B., Bush, R., Matsuzaki, Y., Colitti, L., and T. Narten, "Using 127-Bit IPv6 Prefixes on Inter-Router Links", RFC 6164, DOI 10.17487/RFC6164, April 2011, <https://www.rfc-editor.org/info/rfc6164>.

[RFC6177] Narten, T., Huston, G., and L. Roberts, "IPv6 Address Assignment to End Sites", BCP 157, RFC 6177, DOI 10.17487/RFC6177, March 2011, <https://www.rfc-editor.org/info/rfc6177>.

[RFC6296] Wasserman, M. and F. Baker, "IPv6-to-IPv6 Network Prefix Translation", RFC 6296, DOI 10.17487/RFC6296, June 2011, <https://www.rfc-editor.org/info/rfc6296>.

[RFC6437] Amante, S., Carpenter, B., Jiang, S., and J. Rajahalme, "IPv6 Flow Label Specification", RFC 6437, DOI 10.17487/RFC6437, November 2011, <https://www.rfc-editor.org/info/rfc6437>.

[RFC6583] Gashinsky, I., Jaeggli, J., and W. Kumari, "Operational Neighbor Discovery Problems", RFC 6583, DOI 10.17487/RFC6583, March 2012, <https://www.rfc-editor.org/info/rfc6583>.

[RFC6741] Atkinson, RJ. and SN. Bhatti, "Identifier-Locator Network Protocol (ILNP) Engineering Considerations", RFC 6741, DOI 10.17487/RFC6741, November 2012, <https://www.rfc-editor.org/info/rfc6741>.

[RFC6877] Mawatari, M., Kawashima, M., and C. Byrne, "464XLAT: Combination of Stateful and Stateless Translation", RFC 6877, DOI 10.17487/RFC6877, April 2013, <https://www.rfc-editor.org/info/rfc6877>.

[RFC7084] Singh, H., Beebee, W., Donley, C., and B. Stark, "Basic Requirements for IPv6 Customer Edge Routers", RFC 7084, DOI 10.17487/RFC7084, November 2013, <https://www.rfc-editor.org/info/rfc7084>.

[RFC7094] McPherson, D., Oran, D., Thaler, D., and E. Osterweil, "Architectural Considerations of IP Anycast", RFC 7094, DOI 10.17487/RFC7094, January 2014, <https://www.rfc-editor.org/info/rfc7094>.

[RFC7217] Gont, F., "A Method for Generating Semantically Opaque Interface Identifiers with IPv6 Stateless Address Autoconfiguration (SLAAC)", RFC 7217, DOI 10.17487/RFC7217, April 2014, <https://www.rfc-editor.org/info/rfc7217>.

[RFC7278] Byrne, C., Drown, D., and A. Vizdal, "Extending an IPv6 /64 Prefix from a Third Generation Partnership Project (3GPP) Mobile Interface to a LAN Link", RFC 7278, DOI 10.17487/RFC7278, June 2014, <https://www.rfc-editor.org/info/rfc7278>.

[RFC7296] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., and T. Kivinen, "Internet Key Exchange Protocol Version 2 (IKEv2)", STD 79, RFC 7296, DOI 10.17487/RFC7296, October 2014, <https://www.rfc-editor.org/info/rfc7296>.

[RFC7368] Chown, T., Ed., Arkko, J., Brandt, A., Troan, O., and J. Weil, "IPv6 Home Networking Architecture Principles", RFC 7368, DOI 10.17487/RFC7368, October 2014, <https://www.rfc-editor.org/info/rfc7368>.

[RFC7421] Carpenter, B., Ed., Chown, T., Gont, F., Jiang, S., Petrescu, A., and A. Yourtchenko, "Analysis of the 64-bit Boundary in IPv6 Addressing", RFC 7421, DOI 10.17487/RFC7421, January 2015, <https://www.rfc-editor.org/info/rfc7421>.

[RFC7788] Stenberg, M., Barth, S., and P. Pfister, "Home Networking Control Protocol", RFC 7788, DOI 10.17487/RFC7788, April 2016, <https://www.rfc-editor.org/info/rfc7788>.

[RFC8064] Gont, F., Cooper, A., Thaler, D., and W. Liu, "Recommendation on Stable IPv6 Interface Identifiers", RFC 8064, DOI 10.17487/RFC8064, February 2017, <https://www.rfc-editor.org/info/rfc8064>.

[RFC8084] Fairhurst, G., "Network Transport Circuit Breakers", BCP 208, RFC 8084, DOI 10.17487/RFC8084, March 2017, <https://www.rfc-editor.org/info/rfc8084>.

[RFC8156] Mrugalski, T. and K. Kinnear, "DHCPv6 Failover Protocol", RFC 8156, DOI 10.17487/RFC8156, June 2017, <https://www.rfc-editor.org/info/rfc8156>.

[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>.

## 19.2. Informative References

[I-D.ietf-6man-ipv6-address-generation-privacy]
Cooper, A., Gont, F., and D. Thaler, "Privacy Considerations for IPv6 Address Generation Mechanisms", draft-ietf-6man-ipv6-address-generation-privacy-08 (work in progress), September 2015.

[I-D.ietf-opsec-ipv6-host-scanning]
Gont, F. and T. Chown, "Network Reconnaissance in IPv6 Networks", draft-ietf-opsec-ipv6-host-scanning-08 (work in progress), August 2015.

[I-D.shytyi-opsawg-vysm]
Shytyi, D., Beylier, L., and L. Iannone, "A YANG Module for uCPE management.", draft-shytyi-opsawg-vysm-08 (work in progress), May 2020.

[RFC8273] Brzozowski, J. and G. Van de Velde, "Unique IPv6 Prefix per Host", RFC 8273, DOI 10.17487/RFC8273, December 2017, <https://www.rfc-editor.org/info/rfc8273>.

## Appendix A. ChangeLog

The changes are listed in reverse chronological order, most recent changes appearing at the top of the list.

-00: initial version.

## Authors' Addresses

Gyan Mishra
Verizon Inc.

Email: gyan.s.mishra@verizon.com

Alexandre Petrescu
CEA, LIST
CEA Saclay
Gif-sur-Yvette, Ile-de-France 91190
France

Phone: +33169089223
Email: Alexandre.Petrescu@cea.fr

Naveen Kottapalli
Benu Networks
300 Concord Road
Billerica MA 01821
United States of America

Phone: +1 978 223 4700
Email: nkottapalli@benu.net

Dusan Mudric
Ciena
Canada

Phone: +1-613-670-2425
Email: dmudric@ciena.com

Dmytro Shytyi
SFR
Paris
France

Email: dmytro.shytyi@sfr.com