6MAN Working Group
Internet-Draft
Intended status: Informational
Expires: 8 January 2026

G. Mishra
Verizon Inc.
D. Shytyi
6WIND
A. Petrescu
CEA, LIST
N. Kottapalli
D. Mudric
Ciena

7 July 2025

SLAAC Prefixes with Variable Interface ID (IID) Problem Statement
draft-mishra-v6ops-variable-iids-problem-statement-03

Abstract

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 details the 64-bit boundary problem statement.

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 8 January 2026.

Copyright Notice

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. 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 IID standard.

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 an efficient 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:

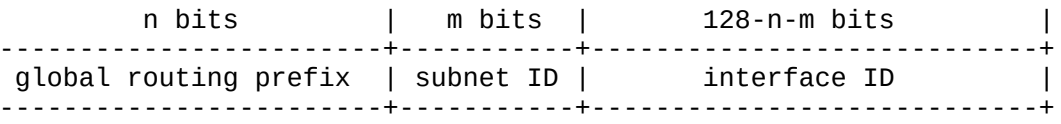

Figure 1: Format of the IPv6 global unicast addresses

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. Problem Statement

This section details the problem statement as to what is broken today with fixed length Stateless Address Autoconfiguration SLAAC [RFC4862] and why it is critical to resolve this problem. The well known Day 1 isse with SLAAC fixed /64 boundary as it exist today is that does not provide direct partiy with other provisioning mechanisms such as Static and DHCPv6 which allows for Variable Length Subnet Mask (VLSM). This has historically been a major problem for deploying DHCPv6 or Static using variable IID due to the incompatibility with SLAAC and thus has shackled Static and DHCPv6 IPv6 provisioning mechanisms to the fixed /64 boundary as well.

The main problem is that SLAAC RA or PD allocates a /64 by the wireless carrier 4G, 5G, 3GPP to mobile handset or hotspot, however segmentation of the /64 via SLAAC is required so that downstream interfaces can be further sub-netted. The use case section of this draft discusses this scenario as one of the use cases for shorter interface identifier, and this use case is the only one stated here in the problem statement as this is broken today with the current SLAAC specification [RFC4862], and there is not any workaround for this use case.

There are two reasons why this was not a problem in the past, but now with increased bandwidth there are more and more devices being piled onto a single handset or mobile hotspot. In the past generations of cellular systems (e.g. 2.5G aka GPRS and some 3G) the bandwidth

available to the User Equipment was not enough to accommodate several applications; bandwidth available was roughly 256Kbit/s. For that reason, users were rarely tempted to use an UE to link other devices than that UE to the Internet. However, with the arrival of 3G, 3G+ (e.g. HSDPA / HSUPA), and even more so with 4G and 4G+, the bandwidth made available to UE increased significantly; this became an average effective of 1Mbit/s and even more. With this available bandwidth, the users are more and more tempted to connect several devices to the Internet. This operation is named 'connection sharing' or 'tethering'. Another answer to this question is that IPv6 technology that is widely used to 'tether' several IP devices to a smartphone is '64share' RFC7278. This technology is used for smartphones but is not so in vehicles. One of the reasons of not being used in vehicles is the lack of scalability: a /64 prefix is shared between the UE ptp link and the subnet (typically Wi-Fi), but can not be further sub-netted to other subnets in the car.

The reason why all devices in a car cannot remain on a single /64 are as follows. These devices have different link-layer technologies, and not all WiFi could be bridged into Ethernet such as to keep all devices into one /64. They could be on links that are not bridgeable: devices on 802.11-OCB cannott be bridged, devices on Bluetooth cannott be bridged, devices on 3GPP cannott be bridged, and so one. Other than the impossibility to bridge several such link-layer technologies there is also a problem of noise: in a vehicle one wants the braking pedal signal to not be disturbed by entertainment sites such as YouTube. That physical technical requirement separation of different link layer technologies segmentation on to different smaller IPv6 subnets cannot be achieved if all devices are on a single /64, or bridged. Therefore, the only possible solution to connect these disparate devices onto a 3GPP network for internet access is to keep these separate link layer technologies segmented onto separate greater then /64 prefix subnets and breaking the /64 boundary that exists today with a Variable IID solution. Thus, when the 3GPP network gives a /64 to the car, and when there are unbridgeable technologies in the car (e.g. WiFi cant be bridged to Bluetooth), then the only possibility is to divide that /64 into two /65s. One /65 would be used on the WiFi and another /65 would be used on Bluetooth. But in order for SLAAC to work with /65 then there is a need to have the shorter interface identifier of length 63. Hence the need of lengths of PIOs other than 64 (variable plen).

There are three scenarios that require SLAAC to be able to be routed between two greater then /64 prefix segments as part of the requirement for variable length IID and what is broken with the current SLAAC specification defined in [RFC4862].

The first scenario is within a car using car manufacturer single SIM for internet access and being able to bridge(Route) other link layer devices like BT via variable IID. In this scenario the communication between downstream devices are all located within the car using the car manufacturer built in SIM card for in-vehicle communication. The in-vehicle scenario covers both the built-in car manufacturer SIM card scenario, or if the car manufacturer does not support built-in SIM card then a single mobile handset providing 3GPP internet access to all devices in the car.

The second scenario is V2V (vehicle to vehicle) between cars requiring SLAAC to subnet the >64 prefix so that the two cars have WiFi connectivity.

This third scenario is a uCPE(Universal Customer Premises Equipment) device is LTE 4G and Wi-Fi capable, and utilizes NFV (Network Function Virtualization) framework, providing SFC (Service Function Chaining), where one VNF (Virtual Network Function) is a CPE Layer 3 router and is the uCPE device which will receive a /64 prefix from 4G 3GPP Wireless provider and would like to be able to provide further segmentation. In order to provide further segmentation and subdivide the /64 into smaller longer prefix subnets variable IID must be employed. In this example we would give 1st /66 to Wi-Fi users, 2nd /66 to Wired connected network device without security, 3rd /66 prefix to VNF firewall instance, and 4th /66 prefix VNF load balancer instance. The uCPE (Universal Customer Premises Equipment) defined in draft [I-D.shytyi-opsawg-vysm].

From a segmented bandwidth perspective while breaking up the /64 subnet into smaller subnets, there is not any impact to the user experience of the now shared bandwidth, as long as the cellular signal has adequate enough bars as far as signal strength to accommodate the now multiple devices sharing the single cellular signal. These scenarios described above are the problems that can only be solved with a variable IID solution. There is no other solution or workaround for this problem.

5. Variable IID Use Cases

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

5.1. SP and Enterprise Customer Use Case

Service Providers and Enterprises want to take advantage of VLSM for Access and Data Center subnets and are not able to do so even when using DHCPv6 or static addressing. The major issue is that VLSM with DHCPv6 and Static addressing is shackled to /64 boundary as well in reality as there will always be at least one or more SLAAC devices on the same subnet. In a real world scenario as you have all threee addressing options available on any subnet, Static, DHCPv6 and SLAAC there is a very high probability that there will be a mix and thus in order for the devices provisioned as DHCPv6 or Static to communicate with any device that is using SLAAC weare now back to the /64 boundary for all devices on any subnet regardless of how the IPv6 address is provisioned.

Opeartors are now stuck with the SLAAC /64 boundary for all subnets across the board and VLSM can never come to fruition with IPv6 even with DHCPv6 or Static address provisioning methods. Unless the standard changes for SLAAC /64 fixed boundary, DHCPv6 and Static will now be bound to the same rules as SLAAC with the /64 mask for DHCPv6 scope and Static addresging.

The SLAAC /64 fixed boundary impacts the proliforation of IPv6 across the board which are failed to start. There have been many complaints over the years with IPv6 as to why we cannot have subnet size less /64. For network designers this makes the case difficult for the move to IPv6 as well as it is very difficult to provide any justification for the /64 boundary other than that is the standard.

Another signficant operator use case is /64 p2p Layer 3 host connections with L2 and L3 isolated networks. In this use case the server clusters and server farms each server compute host CNF, VNF, PNF is how /127 P2P connected to the DC switch fabric. So a single /64 per host so now if you have 65535 CNF, VNF, PNF which with cloud native containerized application workloads such as Kubernetes and RHEL Openshift and Openstack can burn up a single /48 pretty quickly. The amount of address space consumed now grows much faster. Thus the need for much smaller subnet size. In this use case it would even better to be able to have variable SLAAC capable all the way down to /127 for simple host provisioning.

5.2. 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, which appears at the edge of the network. The permission is ’granted’ for end users situated at the edge of the network, and is ’granted’ by advertising a prefix of length 64 inside the PIO option in a RA typically. 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 the device 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 cannot 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).

5.3. Private Networks

Private networks such as Service Provider core not accessible by customers and enterprises where all hosts are trusted are the primary use case for variable IID 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.

5.4. 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 IID 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.

5.5. 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 Wi-Fi endpoint subnets can also now take advantage of variable length IID in deployment scenarios. In cases where multiple routers are deployed in a home environment where routing prefix reachability needs to be advertised where Babel [RFC6126] routing protocol is utilized in those cases variable SLAAC can also be utilized to break up a /64 into multiple smaller subnets.

5.6. 3GPP V2I and V2V networking

In V2I networking (with 3GPP or with IEEE 802.11bd) the IP-OBU in the vehicle receives a /64 prefix from the cellular network (or from a IP-RSU - Road-Side Unit). This /64 prefix can be used to form one address for the egress interface of the Mobile Router (MR, which is also termed 'IP-OBU', for IP On-Board Unit, in IPWAVE WG documents such as RFC8691), but can not be used to form IP addresses for other hosts in the vehicle. In the following two paragraphs we explain this problem.

In certain 3GPP V2I networking use cases a /56 is allocated by the 3GPP infrastructure to the 4G modem of the IP-OBU in the vehicle. In such use case it is possible that the IP-OBU sub-divides the allocated /56 into multiple 'result' /64 prefixes. Such a 'result' /64 prefix could be used to form addresses for deeper subnets in the vehicle, by employing existing SLAAC and existing IPv6-over-foo specifications of Interface ID.

If in other 3GPP V2I networking use-cases the infrastructure does not allocate a /56 (or 'longer' prefix lengths such as a /57, /58../63) to the IP-OBU, i.e. a /64 is allocated to the IP-OBU, then the 'result' prefix obtained after a sub-divide operation can only be of length /65, or /66, or longer. A prefix of such length (longer than 64) can not be used with SLAAC and existing IP-over-foo Interface Identifiers, because the length of all Interface Identifiers in all IPv6-over-foo documents must always be 64, and the length of the IPv6 is always 128bit. The 64bit of an IID added to the 65bit (or more) of a prefix is larger than 128bit. It is for this reason that a SLAAC with other than 64bit Interface IDs (hence a 'Variable Prefix Length IID') is needed.

The problem of /64 allocation to the vehicle is mostly present in V2I use-cases. In V2V use-cases this problem is less apparent but deserves consideration. Until now there was no clearcut design and decision about the infrastructure allocating addresses to several vehicles (just to one, in V2I, see above). In some use-cases, the prefix allocated to one vehicle could be further extended by that vehicle to delegate prefixes to other vehicles nearby which might not have 3GPP connections, but only 802.11-OCB interfaces. In such cases it is again necessary that a /64 allocated by the infrastructure to the first vehicle be further sub-divided in multiple 'result' longer-than-/64 prefixes; and one of these longer-than-64 prefixes might be used for the second vehicle (instead of being used for the internal subnets of the first vehicle); this latter vehicle will need to use a form of IID and IP-over-foo that are not limited by the /64 limit.

5.7. Smart Traffic Lights

Smart traffic lights are traffic lights equipped with a communication system. Smart traffic lights are deployed at intersections of roads and serve the purpose of safely arbitrating the passage of automobiles, pedestrians and cyclists. A typical smart traffic lights setting is made of several computers, included but not limited to: a traffic lights controller, a power controller and a communication gateway. More advanced smart traffic lights are equipped with more computers for radars, detection loops, lidars, V2X wireless capabilities, Wi-Fi, Bluetooth and cellular 4G or 5G. All these computers need to use IP addresses: at least one IP address per computer. Since smart traffic lights are deployed in areas where Internet might not be available by cable, fibre or other Wireless MAN technology the only way to connect all computers in the smart traffic lights setting is to employ a 4G (or 5G) gateway. This gateway obtains typically a /64 prefix from the network operator; there is a problem in subdividing that /64 prefix into smaller prefixes, because the obtained prefixes can not be used by SLAAC, because SLAAC uses Interface IDs of length 64 in practice. Even if the SLAAC specification is independent of the prefix length, the length of the Interface ID dictates the prefix length by side effect (128 minus IID length imposes the prefix length). SLAAC might work with a plen 65 by specification, but all IIDs in all IPv6-over-foo request that IIDs be 64; and the sum of IID len plus plen must be 128.

5.8. 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 IID longer prefixes lengths /65-/128.

5.9. 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 IID subnet could be utilized to reduce the maximum number of hosts on the subnet.

5.10. Permission-less extension of the Network

When one wants to extend the network, one typically wants to add new computers to it. Currently, there are two ways to achieve it: (1) ask the network administrator to provide addresses while also inserting a route towards the new subnet of devices and (2) use NAT. With IPv6, NAT is not desirable. In order to extend the network without asking for permission one needs to obtain addresses and to obtain that route inserted. In order to obtain addresses, one might take advantage of the /64 prefix typically advertised by the network to an edge of it. To do that, one needs to sub-divide the /64 prefix into /65 sub-prefixes (or longer, such as /66, /67, etc.) which could be further advertised in the extension of the network. For the action of inserting a route, the particular topic is outside the scope of this document.

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. 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

* 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

* 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

* 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

* 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.

9. 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.

10. IANA Considerations

No IANA Considerations.

11. Contributors

Brian Carpenter

12. Acknowledgements

13. References

13.1. Normative References

[I-D.bourbaki-6man-classless-ipv6]
Bourbaki, N., "IPv6 is Classless", Work in Progress, Internet-Draft, draft-bourbaki-6man-classless-ipv6-12, 31 March 2025, <https://datatracker.ietf.org/doc/html/draft-bourbaki-6man-classless-ipv6-12>.

[I-D.ietf-6lo-6lobac]
Lynn, K., Martocci, J., Neilson, C., and S. Donaldson, "Transmission of IPv6 over Master-Slave/Token-Passing (MS/TP) Networks", Work in Progress, Internet-Draft, draft-ietf-6lo-6lobac-08, 13 March 2017, <https://datatracker.ietf.org/doc/html/draft-ietf-6lo-6lobac-08>.

[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", Work in Progress, Internet-Draft, draft-ietf-6lowpan-btle-12, 12 February 2013, <https://datatracker.ietf.org/doc/html/draft-ietf-6lowpan-btle-12>.

[I-D.shytyi-opsawg-vysm]
Shytyi, D., Beylier, L., and L. Iannone, "A YANG Module for uCPE management.", Work in Progress, Internet-Draft, draft-shytyi-opsawg-vysm-10, 9 September 2021, <https://datatracker.ietf.org/doc/html/draft-shytyi-opsawg-vysm-10>.

[I-D.templin-aerolink]
Templin, F., "Asymmetric Extended Route Optimization (AERO)", Work in Progress, Internet-Draft, draft-templin-aerolink-82, 10 May 2018, <https://datatracker.ietf.org/doc/html/draft-templin-aerolink-82>.

[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>.

[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>.

[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>.

[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>.

13.2. Informative References

[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

Dmytro Shytyi
6WIND
Paris
France
Email: dmytro@shytyi.net

Alexandre Petrescu
CEA, LIST
CEA Saclay
91190 Gif-sur-Yvette
France
Phone: +33169089223
Email: Alexandre.Petrescu@cea.fr

Naveen Kottapalli
Ciena
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