IPv6 operations
Internet-Draft
Intended status: Informational
Expires: 4 September 2025

N. Buraglio
Energy Sciences Network
O. Caletka
RIPE NCC
J. Linkova
Google

3 March 2025

IPv6-Mostly Networks: Deployment and Operations Considerations
draft-ietf-v6ops-6mops-01

Abstract

This document discusses a deployment scenario called "an IPv6-Mostly network", when IPv6-only and IPv4-enabled endpoints coexist on the same network (network segment, VLAN, SSID etc).

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 4 September 2025.

Table of Contents

1. Introduction 3
2. Requirements Language 4
3. Terminology 4
4. Solution Overview 4
4.1. IPv6-Only Capable Endpoints 5
4.2. IPv6-Only and IPv4-enabled Endpoints Coexistence 5
4.3. Access to IPv4-only Destinations 6
4.3.1. NAT64 6
4.3.2. 464XLAT 7
4.3.3. Signalling NAT64 Prefix to Hosts 7
4.3.4. DNS vs DNS64 7
5. Solution Analysis 8
5.1. IPv6-Only Compared to Dual-Stack 8
5.2. IPv6-Mostly Compared to a Dedicated IPv6-Only Network . . 10
6. Incremental Rollout Considerations 11
6.1. Per-Device and Per-Subnet Incremental Rollout 11
6.2. Rollback Approach 11
6.3. Opt-In and Opt-Out Modes 12
7. Operational Considerations 12
7.1. Address Assignment Policy 12
7.2. Extension Headers 12
7.3. Onlink Communication and Service Discovery 13
7.4. Typical Issues 13
7.4.1. Hosts with Disabled or Disfunctional IPv6 14
7.4.2. Network Extension 14
7.4.3. Multiple Addresses per Device 14
7.4.4. Host Mobility and Renumbering 15
7.4.5. Fragmentation 16
7.4.6. Representing IPv6 Addresses by CLAT 16
7.4.7. IPv4-Dependencies in Network Admission Control . . . 17
7.4.8. Custom DNS Configuration on Endpoints 17
8. Security Considerations 17
9. Privacy Considerations 18
10. IANA Considerations 18
11. References 18
11.1. Normative References 18
11.2. Informative References 19
Acknowledgements 21
Authors' Addresses 21

1. Introduction

While most network operators initially deploy IPv6 alongside their existing IPv4 infrastructure, pure IPv6-only networks remain uncommon outside of the mobile carrier space. This dual-stack approach is seen as a necessary transition phase, allowing operators to gain experience with IPv6 while minimizing disruption.

However, dual-stack networks do not address the core problem driving IPv6 adoption: IPv4 address exhaustion. They still require the same amount of IPv4 resources as IPv4-only networks. Even worse, this dual-stack approach has been demonstrated to be a long-term crutch, frequently masking problems that would otherwise be exposed and remedied as normal operational workflows. Many applications still rely on IPv4, creating a chicken-and-egg problem: IPv6-only networks seem impractical with so many incompatible applications, yet applications continue to rely on IPv4 because IPv6-only networks are rare.

The less control a network operator has over devices and applications, the more difficult it is to break IPv4 dependencies and move to IPv6-only. This is particularly challenging in enterprise networks with legacy IPv4-dependent applications and public Wi-Fi networks where operators cannot guarantee device compatibility.

To enable a gradual transition, operators need to identify which devices can function in IPv6-only mode and which cannot. Creating separate network segments for each type introduces complexity and scalability issues a major hurdle to IPv6-only adoption.

A more desirable approach is to deploy an "IPv6-mostly" network that provides IPv4 on demand. This allows IPv6-capable devices to remain IPv6-only while seamlessly supplying IPv4 to those that require it. An IPv6-mostly network allows Endpoints to operate at the highest level of their network stack evolution, on demand, while still allowing for legacy compatibility.

This document explores the requirements, recommendations, and challenges associated with deploying IPv6-mostly networks in enterprise and public Wi-Fi environments. While the principles discussed may be applicable to other network types, this document’s focus remains on these specific use cases.

2. Requirements Language

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.

3. Terminology

This document reuses most of Terminology section from [RFC8925] .

Endpoint: A device connected to a network and considered a host From the operator’s perspective. However, some endpoint can also extend the network to other physical or logical systems, thereby assuming routing functions. Examples include:

* Corporate laptop: While primarily a host, it might run virtual systems and route traffic to them, extending the network and acting as a router.

* Mobile phone with tethering enabled: Acts as a host on the Wi-Fi network, but also as a router for tethered devices, potentially without the operator’s knowledge or consent.

Network segment: a link (VLAN, a broadcast domain etc) where hosts share the same IP subnet.

PREF64: (or NAT64 prefix): An IPv6 prefix used for IPv6 address synthesis and for network addresses and protocols translation from IPv6 clients to IPv4 servers, [RFC6146].

4. Solution Overview

In a nutshell, an IPv6-mostly network is very much similar to a dual-stack one with two additional key elements:

* The network provides NAT64 ([RFC6146]) functionality, enabling IPv6-only clients to communicate with IPv4-only destinations.

* The DHCPv4 server infrastructure offers DHCPv4 Option 108 as per [RFC8925].

Upon connecting to an IPv6-mostly network segment, an endpoint configures its IP stack based on its capabilities:

* IPv4-Only Endpoint: Acquires an IPv4 address through DHCPv4.

* Dual-Stack Endpoint (Not IPv6-Only Capable): Configures IPv6 addresses using any supported protocol. Additionally, it obtains an IPv4 address via DHCPv4.

* IPv6-only capable endpoint configures its IPv6 addresses and, while performing DHCPv4, includes option 108 ([RFC8925]) into the Parameter Request List. The DHCP server returns the option and, as per [RFC8925] , the endpoint forgoes requesting an IPv4 address, remaining in IPv6-only mode.

An IPv6-mostly network segment can support a mix of IPv4-only, dual-stack, and IPv6-only devices. IPv6-only endpoints utilize the network-provided NAT64 to reach IPv4-only destinations.

The following sections discussed those solution elements in more details.

4.1. IPv6-Only Capable Endpoints

The term "IPv6-only capable endpoint" lacks a strict technical definition. It broadly describes a device that can function without native IPv4 connectivity/IPv4 addresses, providing the same user experience. Examples include but are not limited to:

* An endpoint capable of communicating only with a limited set of endpoints, all of which are available over IPv6.

* An endpoint capable of utilizing only IPv6 sockets, employing NAT64 for accessing IPv4 resources.

* An endpoint implementing a customer-side translator (CLAT) as specified in 464XLAT ([RFC6877]) to provide native IPv4 connectivity for the legacy applications.

4.2. IPv6-Only and IPv4-enabled Endpoints Coexistence

One effective way to restrict IPv4 addresses solely to devices that require them is to enable support for the IPv6-Only Preferred DHCPv4 option (Option 108, [RFC8925]) on the network’s DHCP infrastructure. Currently, most IPv6-only capable systems also support Option 108 and have it enabled by default. By recognizing this option, the network infrastructure can enforce the IPv6-only mode for those devices.

Certain devices, such as resource-constrained embedded systems, may operate in IPv6-only mode without specific adjustments if their communication is limited to IPv6-enabled destinations. Since these systems often lack Option 108 support, administrators may need alternative methods to prevent IPv4 address assignment. One approach is to block IPv4 traffic at the switchport level. This could involve:

* Static ACL: Applying a static filter with a "deny ip any any" rule.

* Dynamic ACL via RADIUS: If 802.1x authentication is in use, RADIUS can provide an ACL blocking all IPv4 traffic.

* Filtering the IPv4 ethertype (0x0800): Some hardware platforms may allow for low level filtering of ethertype 0x0800 preventing IPv4 from passing into or out of a given switchport.

The ACL-based approach has some scalability implications and increases operational complexity, therefore it could only be recommended as a stopgap solution.

4.3. Access to IPv4-only Destinations

4.3.1. NAT64

IPv6-only endpoints require NAT64 to access IPv4-only destinations. Quite often operators choose to combine NAT44 and NAT64 functions, deploying NAT64 at the Internet edge. However, if not all internal services are IPv6-enabled, then NAT64 might need to allow accessing internal destinations as well as external ones. In that case it might be beneficial to deploy NAT64 closer to the users, instead of the network Internet edge, in order to simplify routing/firewall rules and reduce latency.

If IPv4-only internal destinations are using [RFC1918] address space, then the operator MUST NOT use the well-known prefix 64:ff9b::/96 for NAT64 (see section 3.1 of [RFC6052]. In that scenario the operator is required to allocate a network-specific NAT64 prefix from the operator’s address space.).

4.3.2. 464XLAT

Unless the usage of IPv4-only network sockets is prohibited or mitigated in the operating system, enabling CLAT (Customer-side translator) on endpoints is essential for seamless operation of IPv4-only applications in IPv6-only environments. CLAT provides an IPv4 address and IPv4 default route, ensuring functionality even without a native IPv4 address from the network. Without CLAT or similar compatibility layer inside the operating system, IPv4-only applications would fail, negatively impacting user experience and increasing support overhead.

Recommendations for Network Administrators controlling the endpoints:

* CLAT + DHCPv4 Option 108: If the network administrator can control endpoint configuration, CLAT SHOULD be enabled on endpoints sending DHCPv4 Option 108. This streamlines the transition.

* Option 108 Without CLAT MAY be enabled if the administrator is willing to identify and fix IPv4-only systems/applications, or if all applications are confirmed to work in IPv6-only mode.

4.3.3. Signalling NAT64 Prefix to Hosts

Hosts running 464XLAT need to discover the PREF64 (the IPv6 prefix used by NAT64). The network administrator SHOULD configure the first-hop routers to include PREF64 information in Router Advertisements as per ([RFC8781]) even if the network provides DNS64 (so hosts can use DNS64-based prefix discovery, [RFC7050]). [I-D.ietf-v6ops-prefer8781] discusses this recommendation in more details.

4.3.4. DNS vs DNS64

Traditionally, DNS64 (with NAT64) is used to enable IPv6-only endpoints to access IPv4-only destinations. However, using DNS64 has a number of drawbacks, such as:

* DNSSEC Incompatibility: DNS64 responses fail DNSSEC validation.

* Custom Resolvers: Endpoints or applications configured with custom resolvers or running recursive resolvers locally can not benefit from the DNS64 provided by the network. Such systems would need to rely on CLAT or local synthesis instead (Section 7.4.8 discusses this scenario in more details).

* Additional requirements for application: to benefit from DNS64, applications need to be IPv6-enabled, use DNS (do not use IPv4 literals). Many applications do not satisfy those requirements and therefore fail if the endpoint does not have an IPv4 address/ native IPv4 connectivity. Existence of such applications is the main reason why transition to IPv6-only must be incremental (via IPv6-mostly phase) and requires the vast majority of endpoints to support CLAT.

Those concerns make DNS64 is suboptimal and undesirtable solution long-term. To eliminate the needs for DNS64, the network shall signal PREF64 to endpoints (see Section 4.3.3), and the endpoints need to use the obtained PREF64 for performing local synthesis and for CLAT. It should be noted that not every application can benefit from local synthesis performed by the operating system, as it would require the application to use DNS (not IP literals). Therefore DNS64 is only required to support the following cases:

* Systems and applications which perform local synthesis but do not support [RFC8781] for prefix discovery, and can only discover the NAT64 prefix using [RFC7050].

* IPv6-only devices without CLAT (unless those endpoints are guaranteed never to need IPv4-only destinations, e.g. in case of a specialized network segment communicating solely with IPv6-capable destinations).

Using DNS64 however bypasses CLAT for IPv6-capable applications communicating with IPv4-only destinations, making the communication slightly more efficient. For that reason it might be beneficial to run DNS64 even if it is not strictly necessary, provided the the above mentioned drawbacks are adressed.

5. Solution Analysis

5.1. IPv6-Only Compared to Dual-Stack

IPv6-Mostly networks offer significant advantages over traditional dual-stack models where endpoints have both IPv4 and IPv6 addresses:

* Drastically Reduced IPv4 Consumption: Dual-stack deployments don’t solve the core problem of IPv4 address exhaustion. IPv6-Mostly allows to significantly reduce IPv4 consumption, as well as to reclaim IPv4 space. This reduction depends on endpoint capabilities (DHCPv4 Option 108 and CLAT support). In real-world scenarios, like conference Wi-Fi, 60-70% of endpoints may support IPv6-only operation, potentially allowing 2-4 times smaller IPv4 subnets.

* Controlled and incremental phase-out of IPv4: IPv6-Mostly allows for the controlled phase out IPv4 from many endpoints, streamlining operations and improving overall network reliability at a measured and operator-controlled pace.

* Reduced Dependency on DHCPv4: As more devices operate seamlessly in IPv6-only mode, the criticality of DHCPv4 service diminishes significantly. This allows operators to scale down DHCPv4 infrastructure or operate it with less stringent service level objectives (SLOs), optimizing costs and resource allocation.

* Simplified troubleshooting due reduced impact of Happy Eyeballs: when both the client and the server are dual-stack, communication can happen either over IPv6 or over IPv4 and the protocol choice can change over time. This may obscure network issues or make them intermittent, complicating the troubleshooting. In IPv6-Mostly network IPv6-only hosts are much less affected by such an issue. Although modern versions of Happy Eyeballs algorithm support falling back to IPv4 even in case of IPv6-only network, the delay before switching to IPv4 is so long that the fallback does not happen during regular operations. It should be noted, however, that when an IPv6-only client communicates with an IPv4-only destination, the traffic may traverse CLAT or be sent as IPv6 towards the NAT64, depending on how the specific application is written.

The introduction of NAT64 within the IPv6-Mostly model may appear as a drawback, as it inherits many of the limitations associated with NAT44, such as translating flows from different hosts to thesame public IPv4 address, or scalability challenges caused by stateful opeation. However, it’s important to recognize that most IPv4-only or dual-stack networks already rely on NAT44 due to IPv4 address scarcity. For these networks, transitioning to an IPv6-Mostly architecture would simply require enabling NAT64 functionality on existing NAT44 devices.

In a dual-stack network, flows originating from IPv6-only capable endpoints to IPv4-only destinations are, in most cases, translated by NAT44. Within an IPv6-Mostly environment, this role is fulfilled by NAT64. Consequently, the overall load on the NAT infrastructure remains effectively unchanged, with NAT64 essentially replacing NAT44 for a subset of traffic flows.

It is also worth emphasizing that NAT64 is employed solely for facilitating access to IPv4 resources. As the availability of IPv6-enabled resources continues to increase, the volume of traffic traversing NAT64 is expected to diminish proportionally.

As discussed in Section 7.3 coexistence of IPv6-only and IPv4-only hosts on the same link might present challenges for onlink service discovery and onlink peer2peer communications. It might be consider a disadvantge of IPv6-Moslty model for networks where onlink communication between IPv4-only and IPv6-only devices is desirable.

5.2. IPv6-Mostly Compared to a Dedicated IPv6-Only Network

Traditional IPv6-only adoption involves separate networks alongside dual-stack ones. IPv6-Mostly approach offers significant improvements, such as:

* Enhanced Scalability: Separate IPv6-only networks double the number of SSIDs in wireless environments, causing channel congestion and degrading performance. IPv6-Mostly doesn’t require additional SSIDs. Similarly, it allows IPv4 and IPv6-only devices to coexist on the same wired VLANs, eliminating the need of additional layer2 segments, VLAN IDs, and their respective layer3 configurations.

* Operational Simplicity: Managing one network segment for all clients (regardless of IPv4 needs) simplifies operations, improves user experience (no more confusing SSID choices), and reduces support tickets related to mismatched connections. For wired connections dynamic VLAN assignment becomes easier without device-specific IPv6 capability tracking.

* Optimized IPv4 Consumption: User-selected dual-stack networks often lead to unnecessary IPv4 use, as users often connect IPv6-only capable devices to a dual-stack network. IPv6-Mostly network allocates IPv4 addresses only when devices don’t advertise IPv6-only capability (DHCPv4 Option 108).

* Improved Problem Visibility: User-selected fallback to dual-stack networks can mask issues with IPv6-only operation, hindering problem reporting and resolution. IPv6-Mostly forces users to work through any issues, improving identification and enabling fixes for smoother long-term transition.

* Flexible, Incremental Transition: IPv6-Mostly allows for gradual migration on a per-segment or even on per-device basis. Devices become IPv6-only only when deemed fully compatible with that mode.

6. Incremental Rollout Considerations

Migrating endpoints to IPv6-only fundamentally changes network dynamics by removing the IPv4 safety net. This includes the masking effect of traditional Happy Eyeballs algorithm [RFC6555]. IPv6 connectivity issues become far more prominent, including those previously hidden within dual-stack environments. Operators should be prepared to discover and troubleshoot issues in both endpoints and network infrastructure, even if the dual-stack network appeared problem-free.

Some rollout considerations are discussed in the following sections.

6.1. Per-Device and Per-Subnet Incremental Rollout

Limited control over endpoint configuration necessitates a per-subnet rollout, incrementally enabling first PREF64 signalling and then option 108 processing in DHCP. Unless special circumstances like unicast router advertisements for each host, enabling PREF64 signalling will happen per-subnet without per-endpoint opt out mechanism. If endpoint control exists, per-device rollout is possible for option 108 processing (at least for OSes with configurable Option 108). Note that some OSes unconditionally enable Option 108 support, becoming IPv6-only the moment it’s activated on the server side. The following approach is RECOMMENDED:

* PREF64 signalling: Enable PREF64 option for the router advertisement and/or DNS64 (see above). Especially if DNS64 is used, this might affect behaviour of some devices as the path via NAT64 would get generally preferred over native IPv4 path. Rollback at this stage affects the entire subnet.

* DHCP Server-Side Activation: Enable Option 108 processing. Some OSes automatically switch to IPv6-only. Rollback at this stage affects the entire subnet.

* Controlled Endpoint Activation: Enable Option 108 on managed endpoints with per-device rollback possible.

6.2. Rollback Approach

For quick rollback, the administrator SHOULD start with a minimal Option 108 value (300 seconds, Section 3.4 of [RFC8925]) and possibly increase this value as the IPv6-mostly network proves reliable, reducing the likelihood of full-scale rollback. In most deployments, the load of DHCP infrastructure caused by each endpoint restarting the DHCP handshake every 300 seconds would be negligible.

6.3. Opt-In and Opt-Out Modes

Before user-facing deployment, the administrator SHOULD consider a dedicated IPv6-mostly proof-of-concept network for early adopters. While this temporarily sacrifices some IPv6-mostly benefits (Section 5.2), it provides valuable operational experience and early issue detection.

* Opt-In Phase: Invite tech-savvy early adopters to join an IPv6-mostly network and report issues. While response rates may be low, dedicated participants provide valuable troubleshooting data.

* Opt-Out Phase: Incrementally enable PREF64 signalling as well as Option 108. Allow selective disabling for problematic endpoints, in extreme case even by providing a separate network or rolling back IPv6-mostly in that particular subnet. Opting out SHOULD NOT be possible without a problem reports to ensure issue tracking and resolution.

In some scenarios (see Section 7.4.1) the administrator MAY keep a dual-stack network as a last resort fallback mechanism but SHOULD prevent usesrs from connecting to it accidentially (e.g. it should be a hidden SSID with authentication enabled). For recurring temporary networks, for instance during a conference, it might be a good practice to change the network SSID of the last resort fallback network between each event.

7. Operational Considerations

7.1. Address Assignment Policy

As outlined in Section 6.3 of [RFC6877] , CLAT requires either a dedicated IPv6 prefix or, if unavailable, a dedicated IPv6 address. Currently (2024), all implementations use SLAAC for CLAT address acquisition. Therefore, to enable CLAT functionality within IPv6-mostly network segments, first-hop routers MUST be configured to advertise a Prefix Information Option (PIO, [RFC4861]) containing a globally routable SLAAC-suitable prefix with the ’Autonomous Address-Configuration’ (A) flag set to one.

7.2. Extension Headers

Being an IPv6-specific concept, IPv6 extension headers are often neglected or even explictly prohibited by security policies in dual-stack networks. The issues caused by blockig extension headers might be masked by the presense of Happy Eyeballs but become highly visible when there is no IPv4 to fallback to.

The network SHOULD permit at least the following extension headers:

* Fragment Header (Section 4.5 of [RFC8200]).Section 7.4.5 discusses the fragmentation in more details.

* ESP Header, which is used for IPSec traffic, such as VPN and Wi-Fi Calling.

7.3. Onlink Communication and Service Discovery

Shared link is often used for automatic discovery of neighboring devices and their services by means of different protocols like Multicast DNS [RFC6762]. Many devices and/or services are built on an assumption that devices on the same link also share the same set of address families or at least they have one common address family shared between all of them.

In IPv6-Mostly, this assumption is not valid as there might be both IPv4-only and IPv6-only devices on the same link. As per Section 20 of [RFC6762], this means that the link has two unrelated ".local." zones, one for each address family. Discovery of devices across different addres families is impossible, unless some sort of relay is deployed on the link.

This concern, however, is only applicable if onlink communications are desired and permitted by security policies. For networks where peer2peer onlink communications are explicitly denied (it’s often a case for enterprise networks and public WiFi), this issue is not a concern. If onlink peer2peer communication is required, the operator needs to ensure all participating devices are either dual-stack or IPv6-mostly.

7.4. Typical Issues

IPv6-mostly networks expose hidden issues by removing the IPv4 safety net. While implementation bugs vary greatly and are beyond the scope of this document, this section focus on common problems caused by configuration, topology, or design choices. It’s crucial to note that these issues likely pre-exist in dual-stack networks, but remain unnoticed due to IPv4 fallback.

7.4.1. Hosts with Disabled or Disfunctional IPv6

Historically, tech support often advised disabling IPv6 as a quick workaround, leading to devices with disabled IPv6. Similarly, corporate IT may have disabled or filtered IPv6 under the assumption that it's not widely used. Such endpoints requesting Option 108 will fail to connect in an IPv6-mostly network, as they won't receive IPv4 addresses and IPv6 is disabled.

Administrators controlling endpoints SHOULD ensure those endpoints have IPv6 enabled and operational before transitioning the network to IPv6-mostly mode. This includes verifying protocol enablement as well as validation of any central or discreetly managed host based firewalls that could potentially interfere with proper IPv6 function that may be masked by the presence of IPv4, and therefor exp[osed with its removal.

7.4.2. Network Extension

IPv4's NAT44 allows endpoints to extend connectivity to downstream systems without upstream network awareness or permission. This creates challenges in IPv6-mostly deployments where endpoints lack IPv4 addresses:

Solutions and trade-offs:

* Using DHCPv6-PD to allocate prefixes to endpoints ([RFC9663]). Provides downstream systems with IPv6 addresses and native connectivity.

* Enabling the CLAT function on the endpoint. This scenario is similar to the Wireline Network Architecture described in Section 4.1 of [RFC6877] . The downstream systems would receive IPv4 addresses and their IPv4 traffic would be translated to IPv6 by the endpoint. However this approach leads to the downstream systems using IPv4 only and not benefiting from end2end IPv6 connectivity. To enable IPv6 benefits, combine this with IPv6 Prefix Delegation as discussed above.

* Bridging and ND Proxy: The endpoint bridges IPv6 traffic and masks downstream devices behind its MAC address. This can lead to scalability issues (Section 7.4.3) due to the single MAC being mapped to many IPv6 addresses.

7.4.3. Multiple Addresses per Device

Unlike IPv4, where endpoints typically have a single IPv4 address per interface, IPv6 endpoints inherently use multiple addresses:

* Link-local address.

* Temporary address (common on mobile devices for privacy)

* Stable address (for long-term identification)

* CLAT address (in IPv6-mostly/IPv6-only networks)

Endpoints with containers, namespaces, or ND proxy functions may have even more addresses. This poses challenges for network infrastructure devices (SAVI switches, wireless access points, etc.) that map MAC addresses to IPv6 addresses, often with limits to prevent resource exhaustion or DoS attacks. When the number of IPs per MAC limit is exceeded, infastructure devices behavior varies across implementations, leading to inconsistent connectivity loss and other unexpected behavior: while some systems drop new addresses, others delete older entries, causing previously functional addresses to lose connectivity. In all those cases Endpoints and applications don’t receive explicit signalling about the address becoming unusable.

While allocating prefixes to endpoints via DHCP-PD ([RFC9663]) alows to eliminate the issue and corresponding scalability concerns, that solution migth not be supported by all endpoints. The network administrator SHOULD ensure that the deployed network infastructure devices allow sufficient number of IPv6 addresses to be mapped to a client’s MAC and SHOULD monitor for events, indicating that the limit has been reached (such as syslog messages, etc).

7.4.4. Host Mobility and Renumbering

Networks employing dynamic VLAN assignment (e.g., based on 802.1x or MAC-based authentication) can cause endpoints to move between VLANs and IPv6 subnets. As client operating systems do not always handle changes in link-layer state (e.g., VLAN changes) correctly, this mobility often leads to inconsistent IP stack behavior on operating systems, resulting in the persistence of old subnet addresses and potential connectivity issues due to incorrect source address selection. In such networks, the administrator SHOULD configure the link-local addresses of the first-hop routers (such as link-local addresses assigned to router interfaces or the corresponding VRRPv3 (Virtual Router Redundancy Protocol, [RFC9568]) virtual link-local IP address) to be unique within the mobility domain (the set of links the host can move between). For example, if there are two VLANs using subnets 2001:db8:1:a::/64 and 2001:db8:1:b::/64, the administrator might configure the virtual router link-local addresses as ’fe80::2001:db8:1:a’ and ’fe80::2001:db8:1:b’, respectively. [I-D.link-6man-gulla] provides further analysis and discusses this

approach in more detail.

7.4.5. Fragmentation

As the basic IPv6 header is 20 bytes longer than the IPv4 header, transating from IPv4 to IPv6 can result in packets exceeding the path MTU on the IPv6 side. In that case NAT64 creates IPv6 packets with the Fragment Header (see Section 4 of [RFC6145] for more details. As per [RFC6145] , by default the translator fragments IPv4 packets so that they fit in 1280-byte IPv6 packets. It means that all IPv4 packets larger than 1260 bytes are fragmented (or dropped if the DF bit is set).

Administrators SHOULD maximize the path MTU on the IPv6 side (from the translator to IPv6-only hosts) to minimize fragmentation. NAT64 devices SHOULD be configured to use the actual path MTU on the IPv6 side when fragmenting IPv4 packets.

Another common case of IPv6 fragmentation is the use of protocols like DNS and RADIUS, where the server response needs to be sent as a single UDP datagram. Network security policies MUST allow IPv6 fragments for permitted UDP traffic (e.g., DNS, RADIUS) where single-datagram responses are required. Allowing IPv6 fragments for permitted TCP traffic is RECOMMENDED unless the network infrastructure reliably performs TCP MSS clamping.

7.4.6. Representing IPv6 Addresses by CLAT

Certain CLAT implementations face challenges when translating incoming IPv6 packets with native (non-synthesized) source addresses (e.g. ICMPv6 packets sent by intermediate hops on the path). This lack of standardized translation mechanisms can lead to:

* Incomplete Traceroute: Omission of IPv6-only hops between the endpoint and NAT64 translator, hindering troubleshooting.

* Path MTU Discovery Issues: Potential disruptions in the PMTU discovery process.

[I-D.equinox-v6ops-icmpext-xlat-v6only-source] proposes a solution for signalling the actual non-synthesized IPv6 source address while translating ICMPv6 error messages.

7.4.7. IPv4-Dependencies in Network Admission Control

Certain layer 2 network devices (e.g., wireless access points, controllers) may enforce a policy where a host’s network access is contingent upon successful DHCP IPv4 address assignment or tied to results of DHCP snooping. Consequently, hosts supporting Option 108 may experience network access denial or disconnection, as they are not assigned IPv4 addresses.

7.4.8. Custom DNS Configuration on Endpoints

In IPv6-mostly networks without PREF64 in RAs, hosts rely on DNS64 ([RFC7050] to discover the NAT64 prefix for CLAT operation. [RFC8880] requires that queries for AAAA resource records of "ipv4only.arpa." MUST be sent to the recursive resolvers provided by the network, not to the resolvers configured manually, local recursive resolvers etc. However some implementations do not comply with [RFC8880] and, when configured with custom DNS resolvers (e.g., public or corporate DNS) may bypass the network-provided DNS64, preventing NAT64 prefix discovery and hindering CLAT functionality.

Where feasible, administrators SHOULD include PREF64 in RAs within IPv6-mostly networks to minimize reliance on DNS64. Administrators need to be aware of the potential for CLAT failures when endpoints use custom resolvers in environments lacking PREF64.

8. Security Considerations

The proposed deployment scenario inherits security considerations of IPv6 (see [RFC9099]). As IPv6-only device can not fall back to IPv4 if IPv6 connectivity is not available or impacted by a malicious actor. Therefore any attack affecting IPv6 connectivity would have much more drastic outcome, comparing to dual-stack networks.

By signalling a rogue NAT64 prefix to a host, a malicious actor can:

* cause a DoS attack, if the network does not provide NAT64 functions (for that prefix or at all);

* implement MitM attack by intercepting traffic to the rogue prefix.

To countermeasure this attack vector, the network administrators SHOULD configure the first-hop routers to include PREF64 information in Router Advertisements as per [RFC8781] and ensure that layer 2 security measures such as RA-Guard ([RFC6105]) are in place. Section 7 of [RFC8781] discusses this topic in more details. Additionally, [I-D.ietf-v6ops-claton] discusses security implications of enabling CLAT if native IPv4 connectivity is available and recommends disabling CLAT in that case.

Security considerations of using DHCP Option 108 are documented in Section 6 of [RFC8925].

9. Privacy Considerations

This document does not introduce any new privacy considerations. IPv6-mostly networks have the same privacy considerations as other Dual-stacked or IPv6-only networks.

10. IANA Considerations

This memo does not introduce any requests to IANA.

11. References

11.1. Normative References

[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>.

[RFC6105] Levy-Abegnoli, E., Van de Velde, G., Popoviciu, C., and J. Mohacsi, "IPv6 Router Advertisement Guard", RFC 6105, DOI 10.17487/RFC6105, February 2011, <https://www.rfc-editor.org/info/rfc6105>.

[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>.

[RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC 6333, DOI 10.17487/RFC6333, August 2011, <https://www.rfc-editor.org/info/rfc6333>.

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

[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>.

[RFC7050] Savolainen, T., Korhonen, J., and D. Wing, "Discovery of the IPv6 Prefix Used for IPv6 Address Synthesis", RFC 7050, DOI 10.17487/RFC7050, November 2013, <https://www.rfc-editor.org/info/rfc7050>.

[RFC7335] Byrne, C., "IPv4 Service Continuity Prefix", RFC 7335, DOI 10.17487/RFC7335, August 2014, <https://www.rfc-editor.org/info/rfc7335>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

[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>.

[RFC8781] Colitti, L. and J. Linkova, "Discovering PREF64 in Router Advertisements", RFC 8781, DOI 10.17487/RFC8781, April 2020, <https://www.rfc-editor.org/info/rfc8781>.

[RFC8585] Palet Martinez, J., Liu, H. M.-H., and M. Kawashima, "Requirements for IPv6 Customer Edge Routers to Support IPv4-as-a-Service", RFC 8585, DOI 10.17487/RFC8585, May 2019, <https://www.rfc-editor.org/info/rfc8585>.

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

[RFC9099] Vyncke, ., Chittimaneni, K., Kaeo, M., and E. Rey, "Operational Security Considerations for IPv6 Networks", RFC 9099, DOI 10.17487/RFC9099, August 2021, <https://www.rfc-editor.org/info/rfc9099>.

11.2. Informative References

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

[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>.

[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>.

[RFC6145] Li, X., Bao, C., and F. Baker, "IP/ICMP Translation Algorithm", RFC 6145, DOI 10.17487/RFC6145, April 2011, <https://www.rfc-editor.org/info/rfc6145>.

[RFC6762] Cheshire, S. and M. Krochmal, "Multicast DNS", RFC 6762, DOI 10.17487/RFC6762, February 2013, <https://www.rfc-editor.org/info/rfc6762>.

[RFC7225] Boucadair, M., "Discovering NAT64 IPv6 Prefixes Using the Port Control Protocol (PCP)", RFC 7225, DOI 10.17487/RFC7225, May 2014, <https://www.rfc-editor.org/info/rfc7225>.

[RFC8880] Cheshire, S. and D. Schinazi, "Special Use Domain Name 'ipv4only.arpa'", RFC 8880, DOI 10.17487/RFC8880, August 2020, <https://www.rfc-editor.org/info/rfc8880>.

[RFC9568] Lindem, A. and A. Dogra, "Virtual Router Redundancy Protocol (VRRP) Version 3 for IPv4 and IPv6", RFC 9568, DOI 10.17487/RFC9568, April 2024, <https://www.rfc-editor.org/info/rfc9568>.

[RFC9663] Colitti, L., Linkova, J., Ed., and X. Ma, Ed., "Using DHCPv6 Prefix Delegation (DHCPv6-PD) to Allocate Unique IPv6 Prefixes per Client in Large Broadcast Networks", RFC 9663, DOI 10.17487/RFC9663, October 2024, <https://www.rfc-editor.org/info/rfc9663>.

[I-D.ietf-v6ops-claton]
Linkova, J. and T. Jensen, "464XLAT Customer-side Translator (CLAT): Node Recommendations", Work in Progress, Internet-Draft, draft-ietf-v6ops-claton-03, 23 February 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-v6ops-claton-03>.

[I-D.ietf-v6ops-prefer8781]
Buraglio, N., Jensen, T., and J. Linkova, "Prefer use of RFC8781 for Discovery of IPv6 Prefix Used for IPv6 Address Synthesis", Work in Progress, Internet-Draft, draft-ietf-v6ops-prefer8781-01, 25 February 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-v6ops-prefer8781-01>.

[I-D.link-6man-gulla]
Linkova, J., "Using Prefix-Specific Link-Local Addresses to Improve SLAAC Robustness", Work in Progress, Internet-Draft, draft-link-6man-gulla-01, 3 March 2025, <https://datatracker.ietf.org/api/v1/doc/document/draft-link-6man-gulla/>.

[I-D.equinox-v6ops-icmpext-xlat-v6only-source]
Lamparter, D. and J. Linkova, "Using Dummy IPv4 Address and Node Identification Extensions for IP/ICMP translators (XLATs)", Work in Progress, Internet-Draft, draft-equinox-v6ops-icmpext-xlat-v6only-source-01, 23 February 2025, <https://datatracker.ietf.org/doc/html/draft-equinox-v6ops-icmpext-xlat-v6only-source-01>.

Acknowledgements

Thanks to Brian Carpenter, Stuart Cheshire, Lorenzo Colitti, David Farmer, Jordi Palet, Michael Richardson, Matsuzaki Yoshinobu for the discussions, the feedback, and all contribution.

Authors’ Addresses

Nick Buraglio
Energy Sciences Network
IL
United States of America
Email: buraglio@forwardingplane.net, buraglio@es.net

Ondrej Caletka
RIPE NCC
Stationsplein 11
Amsterdam
Netherlands
Email: Ondrej.Caletka@ripe.net

Jen Linkova
Google
1 Darling Island Rd
Pyrmont NSW 2009
Australia
Email: furry13@gmail.com, furry@google.com