

6lo Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: 26 April 2022

C.G. Gomez  
UPC  
A.M. Minaburo  
Acklio  
October 2021

Transmission of SCHC-compressed packets over IEEE 802.15.4 networks  
draft-gomez-6lo-schc-15dot4-01

Abstract

A framework called Static Context Header Compression and fragmentation (SCHC) has been designed with the primary goal of supporting IPv6 over Low Power Wide Area Network (LPWAN) technologies [RFC8724]. One of the SCHC components is a header compression mechanism. If used properly, SCHC header compression allows a greater compression ratio than that achievable with traditional 6LoWPAN header compression [RFC6282]. For this reason, it may make sense to use SCHC header compression in some 6LoWPAN environments, including IEEE 802.15.4 networks. This document specifies how a SCHC-compressed packet can be carried over IEEE 802.15.4 networks.

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 April 2022.

Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document.

Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	2
2. Terminology . . . . .	3
2.1. Requirements language . . . . .	4
2.2. Background on SCHC . . . . .	4
3. Architecture . . . . .	4
3.1. Network topologies . . . . .	4
3.2. Protocol stack . . . . .	4
4. Frame Format . . . . .	5
4.1. SCHC Dispatch . . . . .	6
4.2. SCHC Header . . . . .	6
4.3. Padding . . . . .	6
5. SCHC compression for IPv6, UDP, and CoAP headers . . . . .	6
5.1. SCHC compression for IPv6 and UDP headers . . . . .	6
5.1.1. Compression of IPv6 addresses . . . . .	7
5.1.2. Compression of UDP ports . . . . .	7
5.2. SCHC compression for CoAP headers . . . . .	7
5.3. Header compression examples . . . . .	8
6. Fragmentation and reassembly . . . . .	8
7. IANA Considerations . . . . .	8
8. Security Considerations . . . . .	8
9. Acknowledgments . . . . .	8
10. References . . . . .	8
10.1. Normative References . . . . .	8
10.2. Informative References . . . . .	10
Authors' Addresses . . . . .	10

## 1. Introduction

RFC 6282 is the main specification for IPv6 over Low power Wireless Personal Area Network (6LoWPAN) IPv6 header compression [RFC6282]. This RFC was designed assuming IEEE 802.15.4 as the layer below the 6LoWPAN adaptation layer, and it has also been reused (with proper adaptations) for IPv6 header compression over many other technologies relatively similar to IEEE 802.15.4 in terms of characteristics such as physical layer bit rate, layer 2 maximum payload size, etc. Examples of such technologies comprise BLE, DECT-ULE, ITU G.9959, MS/TP, NFC, and PLC. RFC 6282 provides additional functionality, such as a mechanism for UDP header compression.

In the best cases, RFC 6282 allows to compress a 40-byte IPv6 header down to a 2-byte compressed header (for link-local interactions) or a 3-byte compressed header (when global IPv6 addresses are used). On the other hand, an RFC 6282 compressed UDP header has a typical size of 4 bytes. Therefore, in advantageous conditions, a 48-byte uncompressed IPv6/UDP header may be compressed down to a 6-byte format (when using link-local addresses) or a 7-byte format (for global interactions) by using RFC 6282.

Recently, a framework called Static Context Header Compression (SCHC) has been designed with the primary goal of supporting IPv6 over Low Power Wide Area Network (LPWAN) technologies [RFC8724]. SCHC comprises header compression and fragmentation functionality tailored to the extraordinary constraints of LPWAN technologies, which are more severe than those exhibited by IEEE 802.15.4 or other relatively similar technologies. SCHC header compression allows a greater compression ratio than that of RFC 6282. If used properly, SCHC allows to compress an IPv6/UDP header down to e.g. a single byte. In addition, SCHC can be used to compress Constrained Application Protocol (CoAP) headers as well [RFC7252][RFC8824], which further increases the achievable performance improvement of using SCHC header compression, since there is no 6LoWPAN header compression defined for CoAP. Therefore, it may make sense to use SCHC header compression in some 6LoWPAN environments [I-D.toutain-6lo-6lo-and-schc], including IEEE 802.15.4 networks, considering its greater efficiency.

If SCHC header compression is added to the panoply of header compression mechanisms used in 6LoWPAN environments, then there is a need to signal when a packet header has been compressed by using SCHC. To this end, the present document specifies a 6LoWPAN Dispatch Type for SCHC header compression [RFC4944].

This document specifies how a SCHC-compressed packet can be carried over IEEE 802.15.4 networks. Note that, as per this document, and while SCHC defines fragmentation mechanisms as well, 6LoWPAN/6Lo fragmentation is used when necessary to transport SCHC-compressed packets over IEEE 802.15.4 networks [RFC4944][RFC8930][RFC8931].

TO-DO: indicate here any specific updates of RFC 8724 for use over IEEE 802.15.4.

## 2. Terminology

## 2.1. 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 BCP14 [RFC2119], [RFC8174], when, and only when, they appear in all capitals, as shown here.

## 2.2. Background on SCHC

The reader is expected to be familiar with the terms and concepts defined in the specification of SCHC (RFC 8724).

## 3. Architecture

### 3.1. Network topologies

IEEE 802.15.4 supports two main network topologies: the star topology, and the peer-to-peer (i.e., mesh) topology.

SCHC has been designed for LPWAN technologies, which are typically based on a star topology where constrained devices (e.g., sensors) communicate with a less constrained, central network gateway [RFC 8376]. However, as stated in [draft-ietf-lpwan-architecture], SCHC is generic and it can also be used in networking environments beyond the ones originally considered for SCHC.

SCHC compression is applicable to both star topology and mesh topology IEEE 802.15.4 networks.

### 3.2. Protocol stack

The traditional 6LoWPAN-based protocol stack for constrained devices (Figure 1, left) places the 6LoWPAN adaptation layer between IPv6 and an underlying technology such as IEEE 802.15.4. Suitable upper layer protocols include CoAP [RFC7252] and UDP. (Note that, while CoAP has also been specified over TCP, and TCP may play a significant role in IoT environments [RFC9006], 6LoWPAN header compression has not been defined for TCP.)

6LoWPAN can be envisioned as a set of two main sublayers, where the upper one provides header compression, while the lower one offers fragmentation.

This document defines an alternative approach for packet header compression over IEEE 802.15.4, which leads to a modified protocol stack (Figure 1, right).

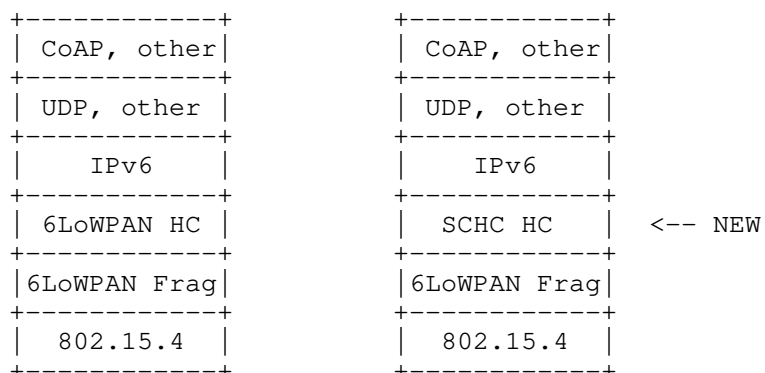


Figure 1: Traditional 6LoWPAN-based protocol stack over IEEE 802.15.4 (left) and alternative protocol stack using SCHC for header compression (right). HC and Frag stand for Header Compression and Fragmentation, respectively.

SCHC header compression may be applied to the headers of different protocols or sets of protocols. Some examples include: i) IPv6 packet headers, ii) joint IPv6 and UDP packet headers, iii) joint IPv6, UDP and CoAP packet headers, etc.

#### 4. Frame Format

This document defines the frame format to be used when a SCHC-compressed packet is carried over IEEE 802.15.4. Such format is carried as IEEE 802.15.4 frame payload. The format comprises a SCHC Dispatch Type, a SCHC Packet (i.e. a SCHC-compressed packet (RFC 8724), and Padding bits, if any). Figure 2 illustrates the described frame format.

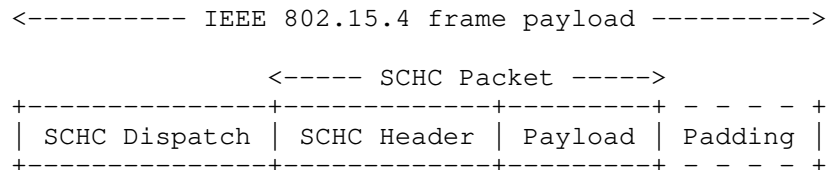


Figure 2: Encapsulated, SCHC-compressed packet. Padding bits are added if needed.

#### 4.1. SCHC Dispatch

Adding SCHC header compression to the panoply of header compression mechanisms used in 6LoWPAN/6Lo environments creates the need to signal when a packet header has been compressed by using SCHC. To this end, the present document specifies the SCHC Dispatch. The SCHC Dispatch indicates that the next field in the frame format is a SCHC-compressed header (SCHC Header in Figure 2, see 4.2)).

This document defines the SCHC Dispatch as a 6LoWPAN Dispatch Type for SCHC header compression [RFC4944]. With the aim to minimize overhead, the present document allocates a 1-byte pattern in Page 0 [RFC8025] for the SCHC Dispatch Type:

SCHC Dispatch Type bit pattern: 01000100 (Page 0) (Note: to be confirmed by IANA))

#### 4.2. SCHC Header

SCHC Header (Figure 2) corresponds to a packet header that has been compressed by using SCHC. As defined in [RFC8724], the SCHC Header comprises a RuleID, and a compression residue. The present specification defines a RuleID size of 8 bits.

#### 4.3. Padding

If SCHC header compression leads to a SCHC Packet size of a non-integer number of bytes, padding bits of value equal to zero MUST be appended to the SCHC Packet as appropriate to align to an octet boundary.

### 5. SCHC compression for IPv6, UDP, and CoAP headers

SCHC header compression may be applied to the headers of different protocols or sets of protocols. Some examples include: i) IPv6 packet headers, ii) joint IPv6 and UDP packet headers, iii) joint IPv6, UDP and CoAP packet headers, etc.

#### 5.1. SCHC compression for IPv6 and UDP headers

With the exception of IPv6 addresses and UDP ports, IPv6 and UDP header fields MUST be compressed as per Section 10 of RFC 8724.

IPv6 addresses are split into two 64-bit-long fields; one for the prefix and one for the Interface Identifier (IID).

To allow for a single Rule being used for both directions, RFC 8724 identifies IPv6 addresses and UDP ports by their role (Dev or App) and not by their position in the header (source or destination). However, such roles are not applicable in some types of 6LoWPAN environments (e.g., when a sender and its destination are both nodes in a mesh topology network). In such cases, the terms Uplink and Downlink as they have been defined in RFC 8724 are not applicable either.

The present specification identifies IPv6 addresses and UDP ports by their position in the header (source or destination). Accordingly, the present specification defines two new values for the Direction Indicator: Transmit (Tx) and Receive (Rx).

#### 5.1.1. Compression of IPv6 addresses

Compression of IPv6 source and destination prefixes MUST be performed as per Section 10.7.1 of RFC 8724.

If the source or destination IID are based on an L2 address, then the IID can be reconstructed with information coming from the L2 header. In that case, the TV is not set, the MO is set to "ignore" and the CDA is set to compute-IID.

As described in [RFC8065], it may be undesirable to build the source IPv6 IID of a device out of the device address. Another static value is used instead. In that case, the TV contains the static value, the MO operator is set to "equal" and the CDA is set to "not-sent".

If several IIDs are possible, then the TV contains the list of possible IIDs, the MO is set to "match-mapping" and the CDA is set to "mapping-sent".

It may also happen that the IID variability only expresses itself on a few bytes. In that case, the TV is set to the stable part of the IID, the MO is set to "MSB" and the CDA is set to "LSB".

#### 5.1.2. Compression of UDP ports

TO-DO

#### 5.2. SCHC compression for CoAP headers

CoAP header fields MUST be compressed as per Sections 4 to 6 of RFC 8824.

### 5.3. Header compression examples

TO-DO: provide examples for IPv6-only, IPv6/UDP and IPv6/UDP/CoAP.

## 6. Fragmentation and reassembly

After applying SCHC header compression to a packet intended for transmission, if the size of the resulting frame format (Section 4) exceeds the IEEE 802.15.4 frame payload space available, such frame format MUST be fragmented, carried and reassembled by means of 6LoWPAN fragmentation and reassembly [RFC4944][RFC8930][RFC8931].

## 7. IANA Considerations

This document requests the allocation of the Dispatch Type Field bit pattern 01000100 (Page 0) as SCHC Dispatch Type.

## 8. Security Considerations

This document does not define SCHC header compression functionality beyond the one defined in RFC 8724. Therefore, the security considerations in section 12.1 of RFC 8724 apply.

As a safety measure, a SCHC decompressor implementing the present specification MUST NOT reconstruct a packet larger than 1500 bytes [RFC8724].

## 9. Acknowledgments

Ana Minaburo and Laurent Toutain suggested for the first time the use of SCHC in environments where 6LoWPAN has traditionally been used. Laurent Toutain, Pascal Thubert, Dominique Barthel, and Guangpeng Li made comments that helped shape this document.

Carles Gomez has been funded in part by the Spanish Government through project PID2019-106808RA-I00, and by Secretaria d'Universitats i Recerca del Departament d'Empresa i Coneixement de la Generalitat de Catalunya 2017 through grant SGR 376.

## 10. References

### 10.1. 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>>.



- [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>>.
- [RFC6282] Hui, J., Ed. and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks", RFC 6282, DOI 10.17487/RFC6282, September 2011, <<https://www.rfc-editor.org/info/rfc6282>>.
- [RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", RFC 7252, DOI 10.17487/RFC7252, June 2014, <<https://www.rfc-editor.org/info/rfc7252>>.
- [RFC8025] Thubert, P., Ed. and R. Cragie, "IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Paging Dispatch", RFC 8025, DOI 10.17487/RFC8025, November 2016, <<https://www.rfc-editor.org/info/rfc8025>>.
- [RFC8065] Thaler, D., "Privacy Considerations for IPv6 Adaptation-Layer Mechanisms", RFC 8065, DOI 10.17487/RFC8065, February 2017, <<https://www.rfc-editor.org/info/rfc8065>>.
- [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>>.
- [RFC8724] Minaburo, A., Toutain, L., Gomez, C., Barthel, D., and JC. Zuniga, "SCHC: Generic Framework for Static Context Header Compression and Fragmentation", RFC 8724, DOI 10.17487/RFC8724, April 2020, <<https://www.rfc-editor.org/info/rfc8724>>.
- [RFC8824] Minaburo, A., Toutain, L., and R. Andreasen, "Static Context Header Compression (SCHC) for the Constrained Application Protocol (CoAP)", RFC 8824, DOI 10.17487/RFC8824, June 2021, <<https://www.rfc-editor.org/info/rfc8824>>.
- [RFC8930] Watteyne, T., Ed., Thubert, P., Ed., and C. Bormann, "On Forwarding 6LoWPAN Fragments over a Multi-Hop IPv6 Network", RFC 8930, DOI 10.17487/RFC8930, November 2020, <<https://www.rfc-editor.org/info/rfc8930>>.

[RFC8931] Thubert, P., Ed., "IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Selective Fragment Recovery", RFC 8931, DOI 10.17487/RFC8931, November 2020, <<https://www.rfc-editor.org/info/rfc8931>>.

## 10.2. Informative References

[I-D.toutain-6lo-6lo-and-schc] Minaburo, A. and L. Toutain, "Comparison of 6lo and SCHC", Work in Progress, Internet-Draft, draft-toutain-6lo-6lo-and-schc-00, 4 November 2019, <<https://www.ietf.org/archive/id/draft-toutain-6lo-6lo-and-schc-00.txt>>.

[RFC9006] Gomez, C., Crowcroft, J., and M. Scharf, "TCP Usage Guidance in the Internet of Things (IoT)", RFC 9006, DOI 10.17487/RFC9006, March 2021, <<https://www.rfc-editor.org/info/rfc9006>>.

## Authors' Addresses

Carles Gomez  
UPC  
C/Esteve Terradas, 7  
08860 Castelldefels  
Spain  
  
Email: [carlesgo@entel.upc.edu](mailto:carlesgo@entel.upc.edu)

Ana Minaburo  
Acklio  
1137A avenue des Champs Blancs  
35510 Cesson-Sevigne Cedex  
France  
  
Email: [ana@ackl.io](mailto:ana@ackl.io)

6lo  
Internet-Draft  
Updates: 6550, 8505, 9010 (if approved)  
Intended status: Standards Track  
Expires: 25 April 2022

P. Thubert, Ed.  
Cisco Systems  
22 October 2021

IPv6 Neighbor Discovery Multicast Address Listener Registration  
draft-ietf-6lo-multicast-registration-01

Abstract

This document updates RFC 8505 to enable a listener to subscribe to an IPv6 anycast or multicast address; the draft updates RFC 6550 (RPL) to add a new Non-Storing multicast mode and support for anycast addresses. This document also extends RFC 9010 to enable the 6LR to inject the anycast and multicast addresses in RPL.

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 25 April 2022.

Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	2
2. Terminology . . . . .	4
2.1. Requirements Language . . . . .	4
2.2. References . . . . .	4
2.3. Glossary . . . . .	5
3. Overview . . . . .	5
4. Extending RFC 7400 . . . . .	8
5. Updating RFC 6550 . . . . .	9
5.1. Updating MOP 3 . . . . .	9
5.2. New Non-Storing Multicast MOP . . . . .	9
5.3. RPL Anycast Operation . . . . .	10
5.4. New RPL Target Option Flags . . . . .	11
6. Updating RFC 8505 . . . . .	11
6.1. New EARO flag . . . . .	11
6.2. Registering Extensions . . . . .	12
7. Updating RFC 9010 . . . . .	13
8. Deployment considerations . . . . .	14
9. Security Considerations . . . . .	16
10. Backward Compatibility . . . . .	16
11. IANA Considerations . . . . .	16
11.1. New RTO flags . . . . .	17
11.2. New RPL Mode of Operation . . . . .	17
11.3. New EARO flags . . . . .	17
11.4. New 6LoWPAN Capability Bits . . . . .	18
12. Acknowledgments . . . . .	18
13. Normative References . . . . .	18
14. Informative References . . . . .	20
Author's Address . . . . .	22

## 1. Introduction

The design of Low Power and Lossy Networks (LLNs) is generally focused on saving energy, which is the most constrained resource of all. Other design constraints, such as a limited memory capacity, duty cycling of the LLN devices and low-power lossy transmissions, derive from that primary concern. The radio (both transmitting or simply listening) is a major energy drain and the LLN protocols must be adapted to allow the nodes to remain sleeping with the radio turned off at most times.

The "Routing Protocol for Low Power and Lossy Networks" [RFC6550] (RPL) to provide IPv6 [RFC8200] routing services within such constraints. To save signaling and routing state in constrained networks, the RPL routing is only performed along a Destination-Oriented Directed Acyclic Graph (DODAG) that is optimized to reach a Root node, as opposed to along the shortest path between 2 peers, whatever that would mean in each LLN.

This trades the quality of peer-to-peer (P2P) paths for a vastly reduced amount of control traffic and routing state that would be required to operate an any-to-any shortest path protocol. Additionally, broken routes may be fixed lazily and on-demand, based on dataplane inconsistency discovery, which avoids wasting energy in the proactive repair of unused paths.

Section 12 of [RFC6550] details the "Storing Mode of Operation with multicast support" with source-independent multicast routing in RPL.

The classical "IPv6 Neighbor Discovery (IPv6 ND) Protocol" [RFC4861] [RFC4862] was defined for serial links and shared transit media such as Ethernet at a time when broadcast was cheap on those media while memory for neighbor cache was expensive. It was thus designed as a reactive protocol that relies on caching and multicast operations for the Address Discovery (aka Lookup) and Duplicate Address Detection (DAD) of IPv6 unicast addresses. Those multicast operations typically impact every node on-link when at most one is really targeted, which is a waste of energy, and imply that all nodes are awake to hear the request, which is inconsistent with power saving (sleeping) modes.

The original 6LoWPAN ND, "Neighbor Discovery Optimizations for 6LoWPAN networks" [RFC6775], was introduced to avoid the excessive use of multicast messages and enable IPv6 ND for operations over energy-constrained nodes. [RFC6775] changes the classical IPv6 ND model to proactively establish the Neighbor Cache Entry (NCE) associated to the unicast address of a 6LoWPAN Node (6LN) in the a 6LoWPAN Router(s) (6LR) that serves it. To that effect, [RFC6775] defines a new Address Registration Option (ARO) that is placed in unicast Neighbor Solicitation (NS) and Neighbor Advertisement (NA) messages between the 6LN and the 6LR.

"Registration Extensions for 6LoWPAN Neighbor Discovery" [RFC8505] updates [RFC6775] into a generic Address Registration mechanism that can be used to access services such as routing and ND proxy and introduces the Extended Address Registration Option (EARO) for that purpose. This provides a routing-agnostic interface for a host to request that the router injects a unicast IPv6 address in the local routing protocol and provide return reachability for that address.

"Routing for RPL Leaves" [RFC9010] provides the router counterpart of the mechanism for a host that implements [RFC8505] to inject its unicast Unique Local Addresses (ULAs) and Global Unicast Addresses (GUAs) in RPL. But though RPL also provides multicast routing, 6LoWPAN ND supports only the registration of unicast addresses and there is no equivalent of [RFC9010] to specify the 6LR behavior upon the registration of one or more multicast address.

The "Multicast Listener Discovery Version 2 (MLDv2) for IPv6" [RFC3810] enables the router to learn which node listens to which multicast address, but as the classical IPv6 ND protocol, MLD relies on multicasting Queries to all nodes, which is unfit for low power operations. As for IPv6 ND, it makes sense to let the 6LNs control when and how they maintain the state associated to their multicast addresses in the 6LR, e.g., during their own wake time. In the case of a constrained node that already implements [RFC8505] for unicast reachability, it makes sense to extend to that support to register the multicast addresses they listen to.

This specification extends [RFC8505] and [RFC9010] to add the capability for the 6LN to register multicast addresses and for the 6LR to inject them in the RPL multicast support.

## 2. Terminology

### 2.1. 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.

### 2.2. References

This document uses terms and concepts that are discussed in:

- \* "Neighbor Discovery for IP version 6" [RFC4861] and "IPv6 Stateless address Autoconfiguration" [RFC4862],
- \* Neighbor Discovery Optimization for Low-Power and Lossy Networks [RFC6775], as well as
- \* "Registration Extensions for 6LoWPAN Neighbor Discovery" [RFC8505] and
- \* "Using RPI Option Type, Routing Header for Source Routes, and IPv6-in-IPv6 Encapsulation in the RPL Data Plane" [RFC9008].

### 2.3. Glossary

This document uses the following acronyms:

6BBR	6LoWPAN Backbone Router
6BBR	6LoWPAN Border Router
6LN	6LoWPAN Node
6LR	6LoWPAN Router
6CIO	Capability Indication Option
AMC	Address Mapping Confirmation
AMR	Address Mapping Request
ARO	Address Registration Option
DAC	Duplicate Address Confirmation
DAD	Duplicate Address Detection
DAR	Duplicate Address Request
EARO	Extended Address Registration Option
EDAC	Extended Duplicate Address Confirmation
EDAR	Extended Duplicate Address Request
DODAG	Destination-Oriented Directed Acyclic Graph
IR	Ingress Replication
LLN	Low-Power and Lossy Network
NA	Neighbor Advertisement
NCE	Neighbor Cache Entry
ND	Neighbor Discovery
NS	Neighbor Solicitation
ROVR	Registration Ownership Verifier
RTO	RPL Target Option
RA	Router Advertisement
RS	Router Solicitation
TID	Transaction ID
TIO	Transit Information Option

### 3. Overview

[RFC8505] is a pre-requisite to this specification. A node that implements this MUST also implement [RFC8505]. This specification does not introduce a new option; it modifies existing options and updates the associated behaviors to enable the Registration for Multicast Addresses as an extension to [RFC8505].

This specification also extends [RFC6550] and [RFC9010] in the case of a route-over multilink subnet based on the RPL routing protocol, to add multicast ingress replication in Non-Storing Mode and anycast support in both Storing and Non-Storing modes. A 6LR that implements the RPL extensions specified therein MUST also implement [RFC9010].

Figure 1 illustrates the classical situation of an LLN as a single IPv6 Subnet, with a 6LoWPAN Border Router (6LBR) that acts as Root for RPL operations and maintains a registry of the active registrations as an abstract data structure called an Address Registrar for 6LoWPAN ND.

The LLN may be a hub-and-spoke access link such as (Low-Power) Wi-Fi [IEEE Std 802.11] and Bluetooth (Low Energy) [IEEE Std 802.15.1], or a Route-Over LLN such as the Wi-SUN mesh [Wi-SUN] that leverages 6LoWPAN [RFC4919][RFC6282] and RPL [RFC6550] over [IEEE Std 802.15.4].

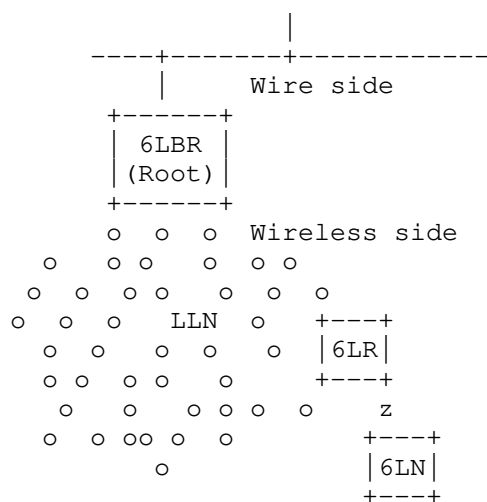


Figure 1: Wireless Mesh

A leaf acting as a 6LN registers its unicast addresses to a RPL router acting as a 6LR, using a unicast NS message with an EARO as specified in [RFC8505]. The registration state is periodically renewed by the Registering Node, before the lifetime indicated in the EARO expires.

With this specification, the 6LNs can now subscribe to the multicast addresses they listen to, using a new M flag in the EARO to signal that the registration is for a multicast address. Multiple 6LN may subscribe to the same multicast address to the same 6LR. Note the use of the term "subscribe": using the EARO registration mechanism, a node registers the unicast addresses that it owns, but subscribes to the multicast addresses that it listens to.



With this specification, the 6LNs can also register the anycast addresses they accept, using a new A flag in the EARO to signal that the registration is for an anycast address. As for multicast, multiple 6LN may register the same anycast address to the same 6LR.

If the R flag is set in the registration of one or more 6LNs for the same address, the 6LR injects the anycast and multicast addresses in RPL, based on the longest registration lifetime across the active registrations for the address.

In the RPL "Storing Mode of Operation with multicast support", the DAO messages for the multicast address percolate along the RPL preferred parent tree and mark a subtree that becomes the multicast tree for that multicast address, with 6LNs that subscribed to the address as the leaves. As prescribed in section 12 of [RFC6550], the 6LR forwards a multicast packet as an individual unicast MAC frame to each peer along the multicast tree, excepting to the node it received the packet from.

In the new RPL "Non-Storing Mode of Operation with multicast support" that is introduced here, the DAO messages announce the multicast addresses as Targets though never as Transit. The multicast distribution is an ingress replication whereby the Root encapsulates the multicast packets to all the 6LRs that are transit for the multicast address, using the same source-routing header as for unicast targets attached to the respective 6LRs.

Broadcasting is typically unreliable in LLNs (no ack) and forces a listener to remain awake, so it generally discouraged. The expectation is thus that in either mode, the 6LRs deliver the multicast packets as individual unicast MAC frames to each of the 6LNs that subscribed to the multicast address.

With this specification, anycast addresses can be injected in RPL in both Storing and Non-Storing modes. In Storing Mode the RPL router accepts DAO from multiple children for the same anycast address, but only forwards a packet to one of the children. In Non-Storing Mode, the Root maintains the list of all the RPL nodes that announced the anycast address as Target, but forwards a given packet to only one of them.

For backward compatibility, this specification allows to build a single DODAG signaled as MOP 1, that conveys anycast, unicast and multicast packets using the same source routing mechanism, more in Section 8.

It is also possible to leverage this specification between the 6LN and the 6LR for the registration of unicast, anycast and multicast IPv6 addresses in networks that are not necessarily LLNs, and/or where the routing protocol between the 6LR and above is not necessarily RPL. In that case, the distribution of packets between the 6LR and the 6LNs may effectively rely on a broadcast or multicast support at the lower layer.

For instance, it is possible to operate a RPL Instance in the new "Non-Storing Mode of Operation with multicast support" (while possibly signaling a MOP of 1) and use "Multicast Protocol for Low-Power and Lossy Networks (MPL)" [RFC7731] for the multicast operation. MPL floods the DODAG with the multicast messages independently of the RPL DODAG topologies. Two variations are possible:

- \* In one possible variation, all the 6LNs set the R flag in the EARO for a multicast target, upon which the 6LRs send a unicast DAO message to the Root; the Root filters out the multicast messages for which there is no listener and only floods when there is.
- \* In a simpler variation, the 6LNs do not set the R flag and the Root floods all the multicast packets over the whole DODAG. Using configuration, it is also possible to control the behavior of the 6LR to ignore the R flag and either always or never send the DAO message, and/or to control the Root and specify which groups it should flood or not flood.

Note that if the configuration instructs the 6LR not to send the DAO, then MPL can really be used in conjunction with RPL Storing Mode as well.

#### 4. Extending RFC 7400

This specification defines a new capability bit for use in the 6CIO as defined by "6LoWPAN-GHC: Generic Header Compression for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)" [RFC7400] and extended in [RFC8505] for use in IPv6 ND messages.

The new "Registration for Multicast Address Supported" (M) flag indicates to the 6LN that the 6LR accepts multicast address registrations as specified in this document and will ensure that packets for the multicast Registered Address will be routed to the 6LNs that registered with the R flag set.

Figure 2 illustrates the M flag in its suggested position (8, counting 0 to 15 in network order in the 16-bit array), to be confirmed by IANA.

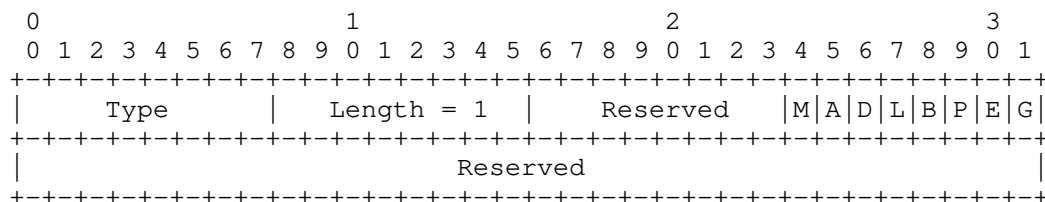


Figure 2: New Capability Bits in the 6CIO

New Option Field:

M 1-bit flag: "Registration for Multicast and Anycast Addresses Supported"

## 5. Updating RFC 6550

### 5.1. Updating MOP 3

RPL supports multicast operations in the "Storing Mode of Operation with multicast support" (MOP 3) which provides source-independent multicast routing in RPL, as prescribed in section 12 of [RFC6550]. MOP 3 is a storing Mode of Operation. This operation builds a multicast tree within the RPL DODAG for each multicast address. This specification provides additional details for the MOP 3 operation.

The expectation in MOP 3 is that the unicast traffic also follows the Storing Mode of Operation. But this is rarely the case in LLN deployments of RPL where the "Non-Storing Mode of Operation" (MOP 1) is the norm. Though it is preferred to build separate RPL Instances, one in MOP 1 and one in MOP 3, this specification allows to hybrid the Storing Mode for multicast and Non-Storing Mode for unicast in the same RPL Instance, more in Section 8.

### 5.2. New Non-Storing Multicast MOP

This specification adds a "Non-Storing Mode of Operation with multicast support" (MOP to be assigned by IANA) whereby the non-storing Mode DAO to the Root may contain multicast addresses in the RPL Target Option (RTO), whereas the Transit Information Option (TIO) can not.

In that mode, the RPL Root performs an ingress replication (IR) operation on the multicast packets, meaning that it transmits one copy of each multicast packet to each 6LR that is a transit for the multicast target, using the same source routing header and encapsulation as it would for a unicast packet for a RPL Unaware Leaf (RUL) attached to that 6LR..

For the intermediate routers, the packet appears as any source routed unicast packet. The difference shows only at the 6LR, that terminates the source routed path and forwards the multicast packet to all 6LNs that registered for the muticast address.

For a packet that is generated by the Root, this means that the Root builds a source routing header as shown in section 8.1.3 of [RFC9008], but for which the last and only the last address is multicast. For a packet that is not generated by the Root, the Root encapsulates the multicast packet as per section 8.2.4 of [RFC9008]. In that case, the outer header is purely unicast, and the encapsulated packet is purely multicast.

For this new mode as well, this specification allows to enable the operation in a MOP 1 brown field, more in Section 8.

### 5.3. RPL Anycast Operation

With multicast, the address has a recognizable format, and a multicast packet is to be delivered to all the active subscribers. In contrast with anycast, the format of the address is not distinguishable from that of unicast. In fact, an external destination (address or prefix) that may be injected from multiple border routers MUST be injected as anycast in RPL.

For either multicast and anycast, there can be multiple registrations from multiple parties, each using a different value of the ROVR field that identifies the individual registration. The 6LR MUST maintain a registration state per value of the ROVR per multicast or anycast address, but inject the route into RPL only once for each address. Since the registrations are considered separate, the check on the TID that acts as registration sequence only applies to the registration with the same ROVR.

The 6LRs that inject multicast and anycast routes into RPL may not be synchronized to advertise same value of the Path Sequence in the RPL TIO. It results that the value the Path Sequence is irrelevant when the target is anycast or multicast, and that it MUST be ignored.

Like the 6LR, a RPL router in Storing Mode propagates the route to its parent(s) in DAO messages once and only once for each address, but it MUST retain a routing table entry for each of the children that advertised the address.

When forwarding multicast packets down the DODAG, the RPL router copies all the children that advertised the address in their DAO messages. In contrast, when forwarding anycast packets down the DODAG, the RPL router **MUST** copy one and only one of the children that advertised the address in their DAO messages, and forward to one parent if there is no such child.

#### 5.4. New RPL Target Option Flags

[RFC6550] recognizes a multicast address by its format (as specified in section 2.7 of [RFC4291]) and applies the specified multicast operation if the address is recognized as multicast. This specification updates [RFC6550] to add the M and A flags to the RTO to indicate that the target address is to be processed as multicast or anycast, respectively.

An RTO that has the M flag set is called a multicast RTO. An RTO that has the A flag set is called an anycast RTO. An RTO that has neither M nor A flag set is called a unicast RTO. The M and A flags are mutually exclusive and **MUST NOT** be both set.

The suggested position for the A and M flags are 2 and 3 counting from 0 to 7 in network order as shown in Figure 3, based on figure 4 of [RFC9010] which defines the flags in position 0 and 1:

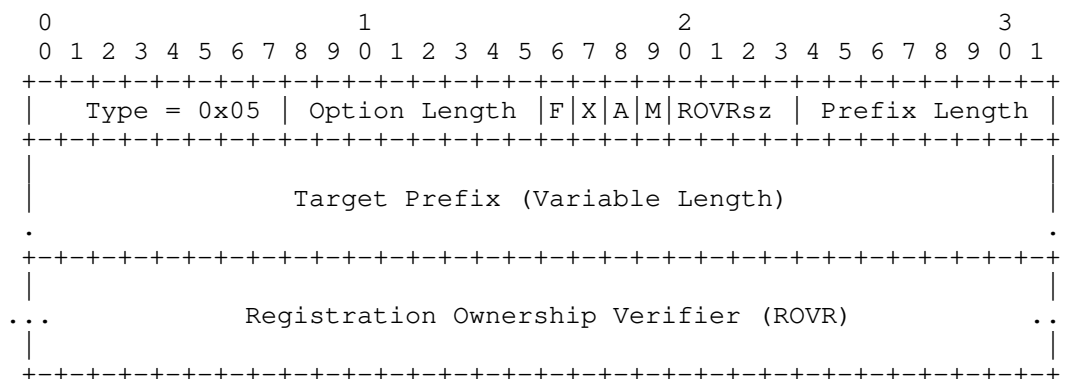


Figure 3: Format of the RPL Target Option

## 6. Updating RFC 8505

### 6.1. New EARO flag

Section 4.1 of [RFC8505] defines the EARO as an extension to the ARO option defined in [RFC6775].

This specification adds a new M flag to the EARO flags field to signal that the Registered Address is a multicast address. When both the M and the R flags are set, the 6LR that conforms to this specification joins the multicast stream, e.g., by injecting the address in the RPL multicast support which is extended in this specification for Non-Storing Mode.

This specification adds a new A flag to the EARO flags field to signal that the Registered Address is an anycast address. When both the A and the R flags are set, the 6LR that conforms to this specification injects the anycast address in the RPL anycast support that is introduced in this specification for both Storing and Non-Storing Modes.

Figure 4 illustrates the A and M flags in their suggested positions (2 and 3, respectively, counting 0 to 7 in network order in the 8-bit array), to be confirmed by IANA.

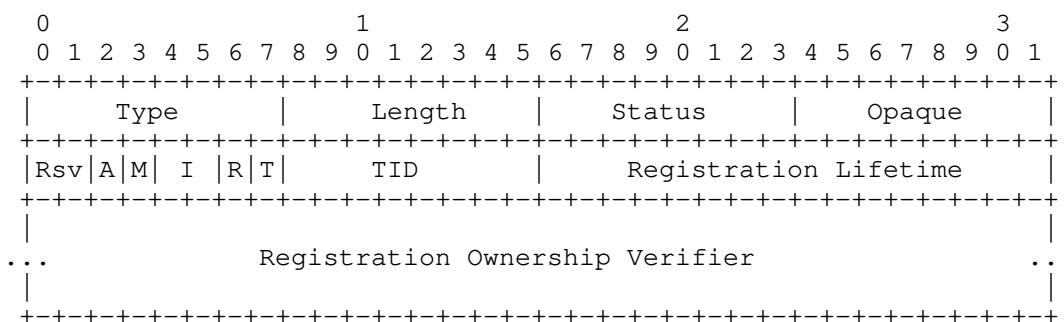


Figure 4: EARO Option Format

New and updated Option Fields:

Rsv 2-bit field: reserved, MUST be set to 0 and ignored by the receiver

A 1-bit flag: "Registration for Anycast Address"

M 1-bit flag: "Registration for Multicast Address"

## 6.2. Registering Extensions

With [RFC8505]:

- \* Only unicast addresses can be registered.
- \* The 6LN must register all its ULA and GUA with a NS(EARO).

- \* The 6LN may set the R flag in the EARO to obtain return reachability services by the 6LR, e.g., through ND proxy operations, or by injecting the route in a route-over subnet.
- \* the 6LR maintains a registration state per Registered Address, including an NCE with the Link Layer Address (LLA) of the Registered Node (the 6LN here).

This specification adds the following behavior:

- \* Registration for multicast and anycast addresses is now supported.
- \* The 6LN MUST also register all the IPv6 multicast addresses that it listens to and it MUST set the M flag in the EARO for those addresses.
- \* The 6LN MAY set the R flag in the EARO to obtain the delivery of the multicast packets by the 6LR, e.g., by MLD proxy operations, or by injecting the address in a route-over subnet or in the Protocol Independent Multicast [RFC7761] protocol.
- \* The 6LN MUST also register all the IPv6 anycast addresses that it supports and it MUST set the A flag in the EARO for those addresses.
- \* The Registration Ownership Verifier (ROVR) in the EARO identifies uniquely a registration within the namespace of the Registered Address. The 6LR MUST maintain a registration state per tuple (IPv6 address, ROVR) for both anycast and multicast types of address, since multiple 6LNs may subscribe to the same address of these types.

## 7. Updating RFC 9010

With [RFC9010]:

- \* The 6LR injects only unicast routes in RPL
- \* upon a registration with the R flag set to 1 in the EARO, the 6LR injects the address in the RPL unicast support.
- \* Upon receiving a packet directed to a unicast address for which it has an active registration, the 6LR delivers the packet as a unicast layer-2 frame to the LLA the nodes that registered the unicast address.

This specification adds the following behavior:

- \* Upon a registration with the R and the M flags set to 1 in the EARO, the 6LR injects the address in the RPL multicast support.
- \* Upon receiving a packet directed to a multicast address for which it has at least one registration, the 6LR delivers a copy of the packet as a unicast layer-2 frame to the LLA of each of the nodes that registered to that multicast address.

## 8. Deployment considerations

With this specification, a RPL DODAG forms a realm, and multiple RPL DODAGs may federated in a single RPL Instance administratively. This means that a multicast address that needs to span a RPL DODAG MUST use a scope of Realm-Local whereas a multicast address that needs to span a RPL Instance MUST use a scope of Admin-Local as discussed in section 3 of "IPv6 Multicast Address Scopes" [RFC7346].

"IPv6 Addressing of IPv4/IPv6 Translators" [RFC6052] enables to embed IPv4 addresses in IPv6 addresses. The Root of a DODAG may leverage that technique to translate IPv4 traffic in IPv6 and route along the RPL domain. When encapsulating an packet with an IPv4 multicast Destination Address, it MUST use form a multicast address and use the appropriate scope, Realm-Local or Admin-Local.

"Unicast-Prefix-based IPv6 Multicast Addresses" [RFC3306] enables to form  $2^{32}$  multicast addresses from a single /64 prefix. If an IPv6 prefix is associated to an Instance or a RPL DODAG, this provides a namespace that can be used in any desired fashion. It is for instance possible for a standard defining organization to form its own registry and allocate 32-bit values from that namespace to network functions or device types. When used within a RPL deployment that is associated with a /64 prefix the IPv6 multicast addresses can be automatically derived from the prefix and the 32-bit value for either a Realm-Local or an Admin-Local multicast address as needed in the configuration.

In a "green field" deployment where all nodes support this specification, it is possible to deploy a single RPL Instance using a multicast MOP for unicast, multicast and anycast addresses.

In a "brown field" where legacy devices that do not support this specification co-exist with upgraded devices, it is RECOMMENDED to deploy one RPL Instance in any Mode of Operation (typically MOP 1) for unicast that legacy nodes can join, and a separate RPL Instance dedicated to multicast and anycast operations using a multicast MOP.



To deploy a Storing Mode multicast operation using MOP 3 in a RPL domain, it is required that there is enough density of RPL routers that support MOP 3 to build a DODAG that covers all the potential listeners and include the spanning multicast trees that are needed to distribute the multicast flows. This might not be the case when extending the capabilities of an existing network.

In the case of the new Non-Storing multicast MOP, arguably the new support is only needed at the 6LRs that will accept multicast listeners. It is still required that each listener can reach at least one such 6LR, so the upgraded 6LRs must be deployed to cover all the 6LN that need multicast services.

Using separate RPL Instances for in the one hand unicast traffic and in the other hand anycast and multicast traffic allows to use different objective function, one favoring the link quality up for unicast collection and one favoring downwards link quality for multicast distribution.

But this might be impractical in some use cases where the signaling and the state to be installed in the devices are very constrained, the upgraded devices are too sparse, or the devices do not support more multiple instances.

When using a single RPL Instance, MOP 3 expects the Storing Mode of Operation for both unicast and multicast, which is an issue in constrained networks that typically use MOP 1 for unicast. This specification allows a mixed mode that is signaled as MOP 1 in the DIO messages for backward compatibility, where limited multicast and/or anycast is available, under the following conditions:

- \* There MUST be enough density of 6LRs that support the mixed mode to cover the all the 6LNs that require multicast or anycast services. In Storing Mode, there MUST be enough density or 6LR that support the mixed mode to also form a DODAG to the Root.
- \* The RPL routers that support the mixed mode and are configured to operate in in accordance with the desired operation in the network.
- \* The MOP signaled in the RPL DODAG Information Object (DIO) messages is MOP 1 to enable the legacy nodes to operate as leaves.
- \* The support of multicast and/or anycast in the RPL Instance SHOULD be signaled by the 6LRs to the 6LN using a 6CIO, see Section 4.

- \* Alternatively, the support of multicast in the RPL domain can be globally known by other means such as configuration or external information such as support of a version of an industry standard that mandates it. In that case, all the routers MUST support the mixed mode.

## 9. Security Considerations

This specification extends [RFC8505], and the security section of that document also applies to this document. In particular, the link layer SHOULD be sufficiently protected to prevent rogue access.

## 10. Backward Compatibility

A legacy 6LN will not register multicast addresses and the service will be the same when the network is upgraded. A legacy 6LR will not set the M flag in the 6CIO and an upgraded 6LN will not register multicast addresses.

As detailed in Section 8, it is possible to add multicast on an existing MOP 1 deployment,

The combination of a multicast address and the M flag set to 0 in an RTO in a MOP 3 RPL Instance is understood by the receiver that supports this specification (the parent) as an indication that the sender (child) does not support this specification, but the RTO is accepted and processed as if the M flag was set for backward compatibility.

When the DODAG is operated in MOP 3, a legacy node will not set the M flag and still expect multicast service as specified in section 12 of [RFC6550]. In MOP 3 an RTO that is received with a target that is multicast and the M bit set to 0 MUST be considered as multicast and MUST be processed as if the M flag is set.

## 11. IANA Considerations

Note to RFC Editor, to be removed: please replace "This RFC" throughout this document by the RFC number for this specification once it is allocated. Also, the I Field is defined in RFC 9010 but we failed to insert it in the subregistry and the flags appear as unspecified though they are.

IANA is requested to make changes under the "Internet Control Message Protocol version 6 (ICMPv6) Parameters" [IANA.ICMP] and the "Routing Protocol for Low Power and Lossy Networks (RPL)" [IANA.RPL] registries, as follows:

### 11.1. New RTO flags

IANA is requested to make additions to the "RPL Target Option Flags" [IANA.RPL.RTO.FLG] subregistry of the "Routing Protocol for Low Power and Lossy Networks (RPL)" registry as indicated in Table 1:

Bit Number	Meaning	Reference
2 (suggested)	A flag: Target is an Anycast Address	This RFC
3 (suggested)	M flag: Target is a Multicast Address	This RFC

Table 1: New RTO flags

### 11.2. New RPL Mode of Operation

IANA is requested to make an addition to the "Mode of Operation" [IANA.RPL.MOP] subregistry of the "Routing Protocol for Low Power and Lossy Networks (RPL)" registry as indicated in Table 2:

Value	Description	Reference
5 (suggested)	Non-Storing Mode of Operation with multicast support	This RFC

Table 2: New RPL Mode of Operation

### 11.3. New EARO flags

IANA is requested to make additions to the "Address Registration Option Flags" [IANA.ICMP.ARO.FLG] of the "Internet Control Message Protocol version 6 (ICMPv6) Parameters" registry as indicated in Table 3:

ARO flag	Meaning	Reference
2 (suggested)	A flag: Registration for Anycast Address	This RFC
3 (suggested)	M flag: Registration for Multicast Address	This RFC
4 and 5	"I" Field	RFC 8505

Table 3: New ARO flags

#### 11.4. New 6LoWPAN Capability Bits

IANA is requested to make an addition to the "6LoWPAN Capability Bits" [IANA.ICMP.6CIO] subregistry subregistry of the "Internet Control Message Protocol version 6 (ICMPv6) Parameters" registry as indicated in Table 4:

Capability Bit	Meaning	Reference
8 (suggested)	M flag: Registration for Multicast and Anycast Addresses Supported	This RFC

Table 4: New 6LoWPAN Capability Bits

## 12. Acknowledgments

## 13. 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>>.
- [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>>.
- [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>>.

- [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>>.
- [RFC6550] Winter, T., Ed., Thubert, P., Ed., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, JP., and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", RFC 6550, DOI 10.17487/RFC6550, March 2012, <<https://www.rfc-editor.org/info/rfc6550>>.
- [RFC6775] Shelby, Z., Ed., Chakrabarti, S., Nordmark, E., and C. Bormann, "Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", RFC 6775, DOI 10.17487/RFC6775, November 2012, <<https://www.rfc-editor.org/info/rfc6775>>.
- [RFC7346] Droms, R., "IPv6 Multicast Address Scopes", RFC 7346, DOI 10.17487/RFC7346, August 2014, <<https://www.rfc-editor.org/info/rfc7346>>.
- [RFC7400] Bormann, C., "6LoWPAN-GHC: Generic Header Compression for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", RFC 7400, DOI 10.17487/RFC7400, November 2014, <<https://www.rfc-editor.org/info/rfc7400>>.
- [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>>.
- [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>>.
- [RFC8505] Thubert, P., Ed., Nordmark, E., Chakrabarti, S., and C. Perkins, "Registration Extensions for IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Neighbor Discovery", RFC 8505, DOI 10.17487/RFC8505, November 2018, <<https://www.rfc-editor.org/info/rfc8505>>.

- [RFC9010] Thubert, P., Ed. and M. Richardson, "Routing for RPL (Routing Protocol for Low-Power and Lossy Networks) Leaves", RFC 9010, DOI 10.17487/RFC9010, April 2021, <<https://www.rfc-editor.org/info/rfc9010>>.
- [IANA.ICMP] IANA, "IANA Registry for ICMPv6", IANA, <https://www.iana.org/assignments/icmpv6-parameters/icmpv6-parameters.xhtml>.
- [IANA.ICMP.ARO.FLG] IANA, "IANA Sub-Registry for the ARO Flags", IANA, <https://www.iana.org/assignments/icmpv6-parameters/icmpv6-parameters.xhtml#icmpv6-adress-registration-option-flags>.
- [IANA.ICMP.6CIO] IANA, "IANA Sub-Registry for the 6LoWPAN Capability Bits", IANA, <https://www.iana.org/assignments/icmpv6-parameters/icmpv6-parameters.xhtml#sixlowpan-capability-bits>.
- [IANA.RPL] IANA, "IANA Registry for the RPL", IANA, <https://www.iana.org/assignments/rpl/rpl.xhtml>.
- [IANA.RPL.RTO.FLG] IANA, "IANA Sub-Registry for the RTO Flags", IANA, <https://www.iana.org/assignments/rpl/rpl.xhtml#rpl-target-option-flags>.
- [IANA.RPL.MOP] IANA, "IANA Sub-Registry for the RPL Mode of Operation", IANA, <https://www.iana.org/assignments/rpl/rpl.xhtml#mop>.

#### 14. Informative References

- [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>>.
- [RFC4919] Kushalnagar, N., Montenegro, G., and C. Schumacher, "IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals", RFC 4919, DOI 10.17487/RFC4919, August 2007, <<https://www.rfc-editor.org/info/rfc4919>>.

- [RFC6282] Hui, J., Ed. and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks", RFC 6282, DOI 10.17487/RFC6282, September 2011, <<https://www.rfc-editor.org/info/rfc6282>>.
- [RFC7731] Hui, J. and R. Kelsey, "Multicast Protocol for Low-Power and Lossy Networks (MPL)", RFC 7731, DOI 10.17487/RFC7731, February 2016, <<https://www.rfc-editor.org/info/rfc7731>>.
- [RFC7761] Fenner, B., Handley, M., Holbrook, H., Kouvelas, I., Parekh, R., Zhang, Z., and L. Zheng, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", STD 83, RFC 7761, DOI 10.17487/RFC7761, March 2016, <<https://www.rfc-editor.org/info/rfc7761>>.
- [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>>.
- [RFC9008] Robles, M.I., Richardson, M., and P. Thubert, "Using RPI Option Type, Routing Header for Source Routes, and IPv6-in-IPv6 Encapsulation in the RPL Data Plane", RFC 9008, DOI 10.17487/RFC9008, April 2021, <<https://www.rfc-editor.org/info/rfc9008>>.
- [Wi-SUN] Heile, B., (Remy), B. L., Zhang, M., and C. E. Perkins, "Wi-SUN FAN Overview", Work in Progress, Internet-Draft, draft-heile-lpwan-wisun-overview-00, 3 July 2017, <<https://datatracker.ietf.org/doc/html/draft-heile-lpwan-wisun-overview-00>>.
- [IEEE Std 802.15.4]  
IEEE standard for Information Technology, "IEEE Std 802.15.4, Part. 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks".
- [IEEE Std 802.11]  
IEEE standard for Information Technology, "IEEE Standard 802.11 - IEEE Standard for Information Technology - Telecommunications and information exchange between systems Local and metropolitan area networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.", <<https://ieeexplore.ieee.org/document/9363693>>.

[IEEE Std 802.15.1]

IEEE standard for Information Technology, "IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements. - Part 15.1: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANs)".

Author's Address

Pascal Thubert (editor)  
Cisco Systems, Inc  
Building D  
45 Allee des Ormes - BP1200  
06254 Mougins - Sophia Antipolis  
France

Phone: +33 497 23 26 34  
Email: pthubert@cisco.com



6Lo Working Group  
Internet-Draft  
Intended status: Experimental  
Expires: 21 April 2022

G. Li  
D. Lou  
L. Iannone  
Huawei  
P. Liu  
China Mobile  
18 October 2021

Native Short Addressing for Low power and Lossy Networks Expansion  
draft-li-6lo-native-short-address-00

Abstract

This document specifies mechanisms of NSA (Native Short Address) that enables IP packet transmission over links where the transmission of a full length address may not be desirable. This document focuses on carrying IP packets across a LLN (Low power and Lossy Network), in which the nodes' location is fixed and changes in the logical topology are caused only by unstable radio connectivity (not physical mobility). The specifications include NSA allocation, routing mechanisms, header format design including length-variable fields, and IPv6 interconnection support.

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 21 April 2022.

Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	2
2. Requirements Notation . . . . .	3
3. Overview . . . . .	4
4. NSA Allocation . . . . .	6
4.1. NSA Addresses and IPv6 Addresses . . . . .	9
4.2. Limitation of Number of Children Node . . . . .	10
5. Routing for a NSA Network . . . . .	10
5.1. Routing toward an NSA endpoint . . . . .	11
5.2. Routing toward an external IPv6 node . . . . .	13
6. Benefits of Native Short Addressing . . . . .	13
7. NSA Header Format . . . . .	14
8. NSA Control Message . . . . .	16
9. IANA Considerations . . . . .	17
9.1. Dispatch Type Field . . . . .	17
9.2. Allocation Function Registry . . . . .	17
9.3. ICMP NSA Control Message . . . . .	18
10. Security Considerations . . . . .	18
11. References . . . . .	18
11.1. Normative References . . . . .	18
11.2. Informative References . . . . .	19
Authors' Addresses . . . . .	20

## 1. Introduction

There is an ongoing massive expansion of the network edge that is driven by the "Internet of Things" (IoT) technology, especially over low-power links which often, in the past, did not support IP packet transmission.

Particularly driven by the requirements stemming from Industry 4.0 and Smart City deployments, more and more devices/things are connected to the Internet. Sensors in plants/parking bays/mines, temperature/humidity/flash sensors in museums, normally are located in a fixed position and are networked by low power and lossy links. Comparing with traditional scenarios, scalability of the (edge) network along with lower power consumption are key technical requirements. Moreover, large-scale Low power Lossy Networks (LLNs)

are expected to be able to carry IPv6 packets over their links, together with an efficient access to native IPv6 domains. These kind of networks, do not necessarily imply nodes mobility [RFC5673], whilst some topology changes may still happen because of fluctuating radio link quality.

The work in [SIXLOWPAN]/[SIXLO]/[LPWAN] Working Groups address many fundamental issues for those type of deployments. Those deployments can be considered an instantiation of what [RFC8799] defines as "limited domains". For instance, the 6lowpan compression technology ([RFC4944] and [RFC6282]) addresses the problem of IPv6 transmission over LLNs, making it possible to interconnect IPv6-based IoT networks and the Internet. [RFC8138] introduces a framework for implementing multi-hop routing on an LLN using a compressed routing header, which works also with RPL (Routing Protocol for LLNs [RFC6550]). This technique enables the ability to forward IPv6 packets within the domain without the need of decompression. In addition, SCHC (Generic Framework for Static Context Header Compression and Fragmentation [RFC8724]) enables even more compression by using a common static context.

The specifications in this document leverage on previous work, namely using the dispatch type field ([RFC4944], [RFC8025]) that allows to accommodate the proposed address format. This means that except the addresses (source and destination) the other fields of the header will be compressed mostly according to LPWAN\_IPHC. The proposed addressing is independent of Unique Local Addresses [RFC4193], which has a dependency on specific link-layer conventions [RFC6282]. It is also different from stateful address allocation that requires all nodes to obtain addresses from a centralized DHCP server, which leads to long network startup time and consumption of extra bandwidth. Compared to RPL-based routing [RFC6550], NSA avoids the extra overhead of address assignment by integrating address assignment and tree forming together. Furthermore, NSA provides shorter packet header than unstoring-mode RPL and much smaller routing table size than storing-mode RPL.

## 2. Requirements Notation

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] and [RFC8174] when, and only when, they appear in all capitals, as shown here.

### 3. Overview

Native Short Address (NSA) is an efficient topology-based network layer address assignment mechanism that is performed in a decentralized fashion. The NSA nodes are aware of its own IPv6 address constructed by IPv6 prefix (by configuration) and NSA (see Section 5.2). Inside the NSA domain, nodes communicate with each other using only NSA addresses. It is a smaller address space compared to the huge IPv6 addressing space. The NSA enables stateless forwarding in most cases. When IPv6 communication occurs between nodes inside the NSA domain and external IPv6 nodes, the border router, which plays as well the role of "root" in the addressing tree, performs network layer translation (as per Section 5.2 and [RFC6282]). The architecture of NSA network is showed in Figure 1.

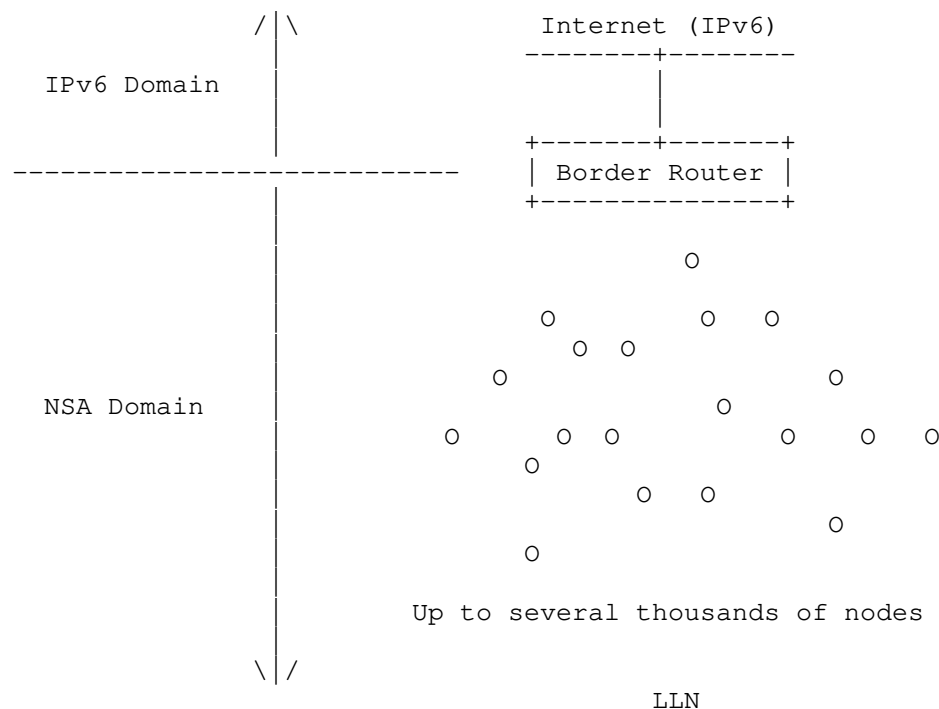


Figure 1: The architecture of general NSA networks.

The overall design objective is centered on how to minimize the packet overhead and reduce the size of routing table to save communication energy in a large-scale IoT LLN network. NSA eliminates compression/decompression of address and also reduces the amount of information synchronization messages, so it actually

reduces computation complexity during packets parsing and forwarding. To this end, NSA uses a context-independent address encoding mechanism. It does not carry any field about address context in the packet. It carries source and destination addresses by variable length fields whose size can be reduced to one octet each in the best case. This allows the NSA packet header to be smaller than LOWPAN\_IPHC's 7 octets (see Figure 2), down to 4 octets, representing around 40% reduction in the header size. Considering that devices in the target limited domain are strongly constrained in resources, while still requiring to use a global unicast IPv6 address to identify them, 7 octets is the smallest size that LOWPAN\_IPHC can achieve in a multi-hop environment, higher than the 2-3 octets necessary in a link-local communication [RFC6282].

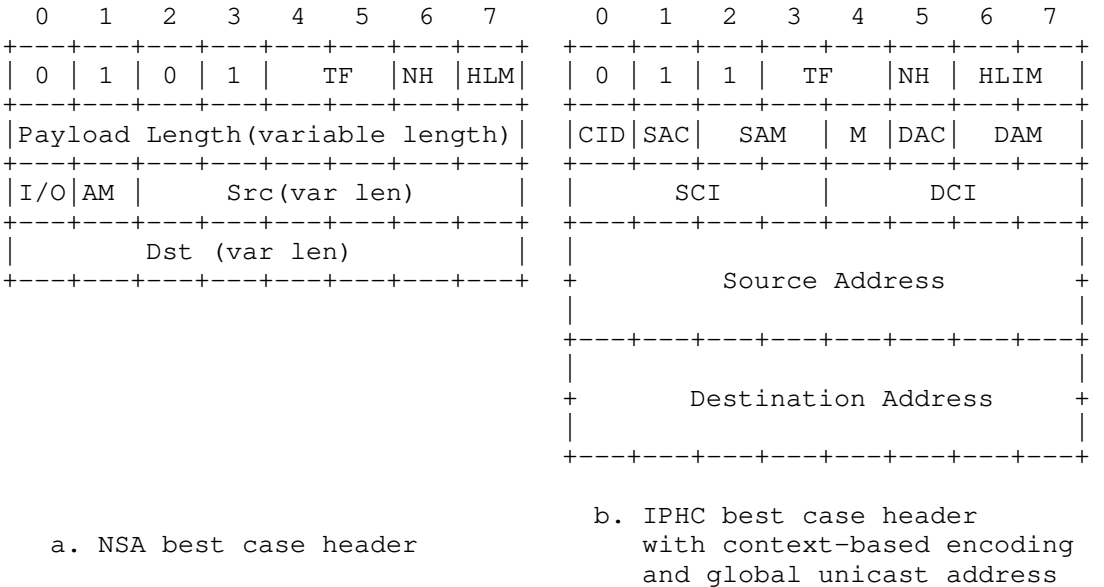


Figure 2: Best case of NSA and LOWPAN\_IPHC packet header.

There are three distinct NSA features that allow NSA to be efficient, namely:

1. Native hort address allocation (see Section 4),
2. Stateless routing (see Section 5),
3. Compact header format design (see Section 7) that avoids context and compression.

#### 4. NSA Allocation

In an NSA network, there are 3 types of roles, namely:

- \* Root,
- \* Forwarder,
- \* Leaf.

The basic rules of allocation include:

- \* Each node's address is prefixed by their parent's address.
- \* The forwarder runs an AF (Allocation Function) to generate its children's addresses.
- \* All nodes run the same AF in the same network instance.

Normally, the root role is assigned to the border router when the LLN bootstraps. An example of a possible result of an NSA deployment is shown in Figure 3.

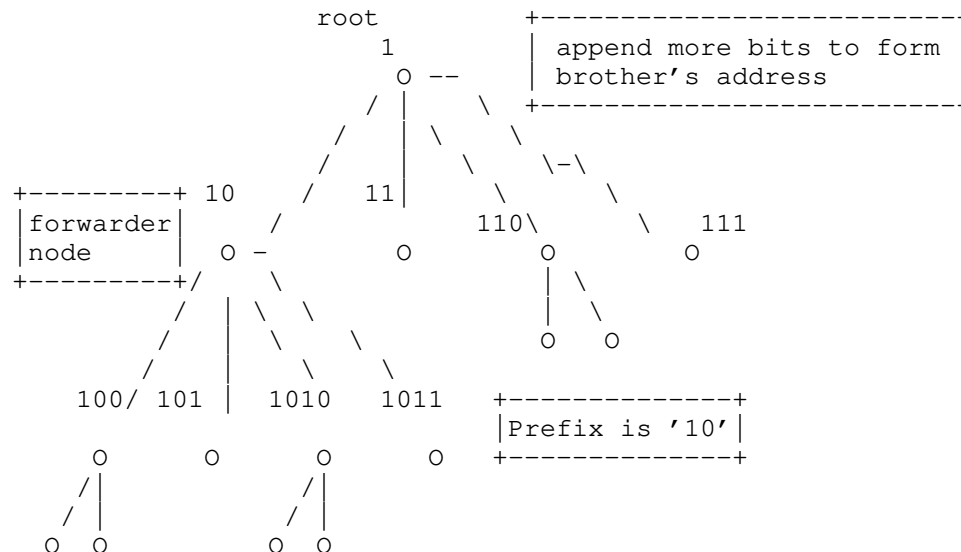


Figure 3: An example of NSA addresses allocation.

Each node acquiring a native short address needs to send an Address Request (AR) message to its link layer neighbors and wait for the response. In the AR message, the node needs to designate a 'role'

value (forwarder or leaf) and the 'nodeid'. Forwarder and Leaf roles can be assigned similarly to IEEE 802.15.4, which distinguishes between Full-Function Devices (FFD) and reduced function devices (RFD) (cf., [ZigBee]). If a neighbor is neither a forwarder nor the root, it will drop the message silently. Otherwise, the neighbor should calculate an address based on parameters in the AR message. After the neighbor node assigned an address for node, it stores the suffix of that address as the interface ID towards the node. Then, it generates and sends Address Assignment (AA) message back. Once a node receives a valid AA response, it uses that assigned address as its own network layer address, thus becomes a child of the address assigner. It will then ignore replies from other neighbors. If a node does not receive any response after an pre-defined interval, it will send the AR message again. It is RECOMMENDED that nodes re-send the AR message up to 3 times, if no answer is received they SHOULD stop.

The allocation function AF(role,i) used in this document is defined in Figure 4. Where every forwarder node should store and maintain two indexes, one for the children that are forwarders and one for the children that are leaves (starting at 0 for the first child in each role). Let's call the first index 'f', as of forwarder, and the second 'l' as for leaves. The '+' symbol indicates a concatenation operation; the b() operation indicates the binary string conversion operation with leading zeros trimmed (note that in this case b(0) is an empty string).

```
AF(role, f, l) = 'address of the node performing the function'
                + (role == leaf? b(l++):b(f++))
                + (role == leaf?'1':'0'),
in which, f and l are the indexes of respectively the forwarders
and the leaves at this layer (starting at 0).
```

Figure 4: Definition of the Allocation Function (AF) of forwarder/root nodes.

Taking the example of the topology in Figure 3, the proposed AF works as follows.

At the top level, there are 4 children of root, two are forwarders and the other two are leaves. Starting from the left most node and moving to the right, the root node applies the AF as follows:

\* For the first child, which is a forwarder:

```
- A('forwarder', 0, 0) = '1'(root address) + b(0) + '0' = '1' +
  '' + '0' = 10
```

- Index f is increased by one and is now equal 1 (f=1)
- \* For the second child, which is a leaf:
  - $A('leaf', 1, 0) = '1'(\text{root address}) + b(0) + '1' = '1' + '' + '1' = 11$
  - Index l is increased by one and is now equal 1 (l=1)
- \* For the third child, which is a forwarder:
  - $A('forwarder', 1, 1) = '1'(\text{root address}) + b(1) + '0' = '1' + '1' + '0' = 110$
  - Index f is increased by one and is now equal 2 (f=2)
- \* For the forth child, which is a leaf:
  - $A('leaf', 2, 1) = '1'(\text{root address}) + b(1) + '1' = '1' + '1' + '1' = 111$
  - Index l is increased by one and is now equal 2 (l=2)

The first level addresses have now been assigned. Let's now have a look how the node 10 (the first forwarder child of the root) applies the same Allocation function. Note that node 10 will use its own 'f' and 'l' indexes initialized to 0. Starting again from the left most node, node 10 applies the AF as follows:

- \* For the first child, which is a forwarder:
  - $A('forwarder', 0, 0) = '10'(\text{node address}) + b(0) + '0' = '10' + '' + '0' = 100$
  - Index f is increased by one and is now equal 1 (f=1)
- \* For the second child, which is a leaf:
  - $A('leaf', 1, 0) = '10'(\text{node address}) + b(0) + '1' = '10' + '' + '1' = 101$
  - Index l is increased by one and is now equal 1 (l=1)
- \* For the third child, which is a forwarder:
  - $A('forwarder', 1, 1) = '10'(\text{node address}) + b(1) + '0' = '10' + '1' + '0' = 1010$



- Index f is increased by one and is now equal 2 (f=2)
- \* For the forth child, which is a leaf:
  - $A('leaf', 2, 1) = '10'(\text{root address}) + b(1) + '1' = '10' + '1' + '1' = 1011$
  - Index l is increased by one and is now equal 2 (l=2)

Note how the children of the same parent all have the same prefix (10 in this example).

The Allocation Function can be different from the one defined in Figure 4, but all nodes know which one to use by configuration. The use of one and only one AF is allowed in an NSA domain. It is RECOMMENDED that implementations support at least the AF proposed in this document (cf. Section 9).

#### 4.1. NSA Addresses and IPv6 Addresses

Obtaining a full IPv6 address from a NSA address is pretty straightforward. First the NSA address is concatenated to the configured IPv6 prefix. Since the length of the NSA address is very likely smaller than 64 bits (the interface ID length in IPv6), the node needs to pad it with zeros ('0') used as most significative bits. The full IPv6 address will look like: IPv6 prefix + "000...000" + NSA (or in IPv6 notation <IPv6 Prefix>::<NSA>). The NSA is assigned by the root/forwarder as previously described.

In an IPv6 communication, the node will derive the NSA address as the short source address from its own IPv6 address by simply removing the IPv6 prefix and all leading zeros before the NSA part. The node will compare the destination IPv6 address with its own IPv6 address. If they have the same prefix, it means that the destination is in the local NSA domain and its corresponding NSA address will be extracted as the short destination address (and the I/O Flag can be set accordingly). Otherwise, it will be a communication towards the Internet. In that case, a mapping mechanism implemented on the root node will generate a short address to be mapped to the full IPv6 destination address. As previously stated, the mapping mechanism is out of the scope of this document.

Since the short mapped address is generated on the root, when the node first open the connection toward the external site, with a first packet, the destination address is set to the full, uncompressed, IPv6 address. Once the packet arrives to the root node, performing the destination address lookup the root will notice that a full IPv6 address is being used and will trigger the short address generation

mechanism and create a new mapping. Such, mapping is communicated to the source node via a new dedicated ICMP message (see Section 8). Once the node originating the communication receives such a message it MUST use the mapped short address for any further communication.

#### 4.2. Limitation of Number of Children Node

The maximum number of child nodes is determined by the specific AF used. IEEE 802.15.5 has explored the use of a per-branch setup, which, however, incurs scalability problems [LEE10]. NSA allocation design is more flexible and extensible than the one proposed in IEEE 802.15.5. The AF used as example in this document does not need any specific setup network by network, though it is still limited by the maximum length of addresses. For the special case of the parent connecting to huge amount of children, a variant of the proposed AF can be designed to fulfill the requirement.

#### 5. Routing for a NSA Network

Internal and external communication in an NSA network works slightly different. For internal communications, among NSA endpoints, packets carry native short addresses and no special operation is needed. For external communications, the root is responsible to perform the translation between native short addresses and IPv6 addresses. For instance, for a packet entering into the NSA domain, the root will extract the native short address of the destination from the suffix of the IPv6 address, by removing all leading '0's. It will also map the source IPv6 address to a mapped native short address, in order to make it more efficient for communication inside the NSA domain.

The root has to store the mapping between external IPv6 addresses and their assigned mapped Native Short Addresses. The method of generating those mapping is out of scope of this document, however, the addressing space for the external NSA has to be maintained separate from the internal NSA address space. Overlap are allowed since the two addressing space are distinguishable in the packets by the use of the I/O field, as explained later on.

The following paragraphs will detail the routing operations for both internal and external communication. The intra-network routing procedure depends on the specific AF used. Here we will use the AF previously introduced (see Figure 4) to illustrate the routing procedure.

### 5.1. Routing toward an NSA endpoint

To perform forwarding operations, NSA nodes access the I/O field in the NSA header (see Section 7). When its value is 1, the packet is destined to a internal NSA node, so it is an inner-domain packet. Otherwise, the packet is destined to an external IPv6 node, so it is called an outer-domain packet. Intra-domain packets carry a native short addresses in the source and the destination address fields. More specifically the destination address field is the address of another node in the same NSA domain. As such an NSA node performs the following sequence of actions, also see Figure 5:

1. Get destination address from packet (abbreviated to DA) and the current node's address (abbreviated to CA). Go to step 2.
2. If length of DA is smaller than length of CA, send the packet to parent node, exit. Otherwise, go to step 3.
3. If length of DA equals to length of CA, go to step 4. Otherwise, go to step 5.
4. If DA and CA are the same, the packet arrived at destination, exit. Otherwise, send the packet to parent node, exit.
5. Check whether CA is equal to the prefix of DA. If yes, go to step 6. Otherwise, send the packet to parent node, exit.
6. Calculate which child is the next hop address and forward packet to it. With the AF propose in this document such operation reduces to reading the DA's bits starting from the position equals to the length of CA, then skip all '1' until the first '0' or the last bit of DA. The sub-string obtained in such a way is the address of direct child of current node.
7. If any exception happens in the above steps, drop the packet and send error notification.

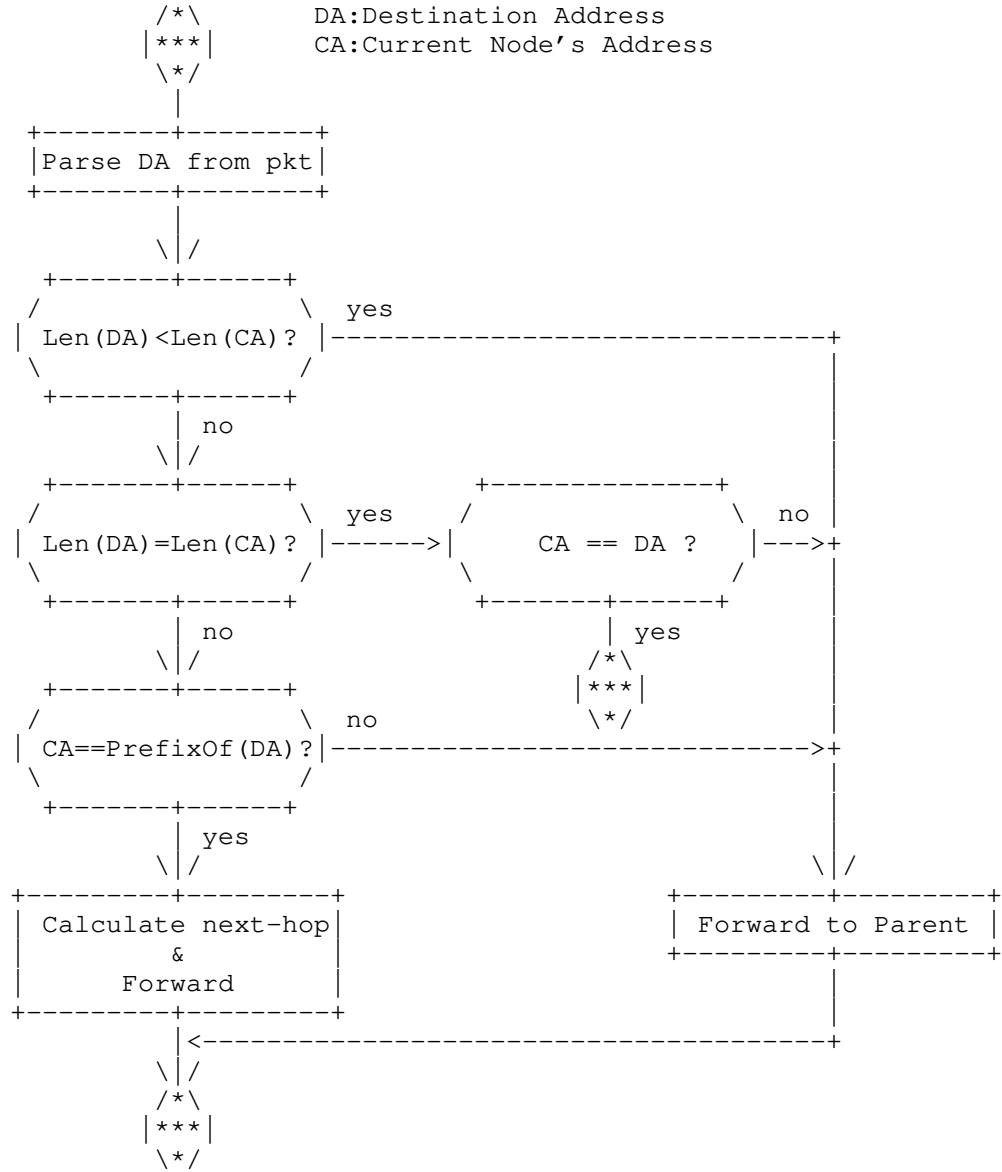


Figure 5: Flow Chart of Internal Routing Procedure

In the case of packet arriving from the Internet (external IPv6 domain toward the local NSA domain) header adaptation operation is performed by the root node. Concerning the destination address, the root builds the native short address of the destination by removing the prefix and the leading '0's of the suffix of the destination

address. Meanwhile, it checks whether it exists already a mapping between the source address and a mapped NSA address to be used as source address in the NSA packet. If not it creates one. Then the root creates the inner-domain packet. It uses the NSA address as destination setting the I/O field to 1 so to route the packet to as described above to the destination node. The mapped NSA address is used as source address and the fact that is a Mapped Address is signaled by setting to 1 the MA field.

## 5.2. Routing toward an external IPv6 node

In the case that the I/O field (cf. Section 7) is set to 0, the packet is destined to an external IPv6 node, it is an outer-domain packet. As such the destination address is either a full IPv6 address (for the first packet of a communication) or a mapped short address generated by the root node and not belonging to any node inside the NSA domain.

All NSA nodes (except root) just send packets that are destined outside the local domain (I/O field equal 0) to their parent, not even looking at the actual destination address. Eventually all packets will reach the root node, which acts as gateway. The Root node is able to map the destination NSA address to the corresponding full IPv6 address. Also, the root node is able to rebuild the full source IPv6 address by concatenating the IPv6 prefix and the NSA address as explained in Section 5.2. Other fields of the header are also decompressed as described in Section 7. A full IPv6 header replaces the original NSA header in the packet, which is then forwarded according to traditional IPv6 protocol.

## 6. Benefits of Native Short Addressing

The NSA use a single set of messages for address assignment and tree forming. It is not more complex than RPL tree forming. So, NSA saves the overhead of address assignment of RPL.

Comparing to RPL with storing mode (see [RFC6550]), there is no need for a NSA node to generate and store routing table entries in the normal case. One of the potential issues is the risk of renumbering of addresses when the topology changes. The topology change could happen in two different scenarios, the high mobility scenario where nodes are moving quite often or even all the time (e.g. UAV) and the unstable link connection scenario where the locations of nodes are fixed, but the connections are broken from time to time. The later is usually called "logic topology change", which covers most of IoT scenarios, see [IoTSurvey].

A "moving" node may possibly be the root of a whole sub-tree, with many children and grand-children, hence, renumbering would introduce a non-negligible cost. Instead of "renumbering", the sub-tree rooted on the "moving" node, its address and the addresses of all its children and grand-children will be kept unchanged. A specific entry in the routing tables of the original and new parent nodes will be created, in order to make the sub-tree still reachable.

Herewith one example, also depicted in Figure Figure 6, node A with the address of 1000 somehow moves from node B (original parent) to node C (new parent). In this case, the routing tables in B, C and their parents' nodes should be updated by adding a new route to "1000", the address of node A. Meanwhile, the original parent (node B) should keep its original address assignment. Comparing with renumbering the addresses of node A and its children, the cost of adding one new route to their parent nodes is much lower, although in this case, the NSA does not implement complete stateless routing.

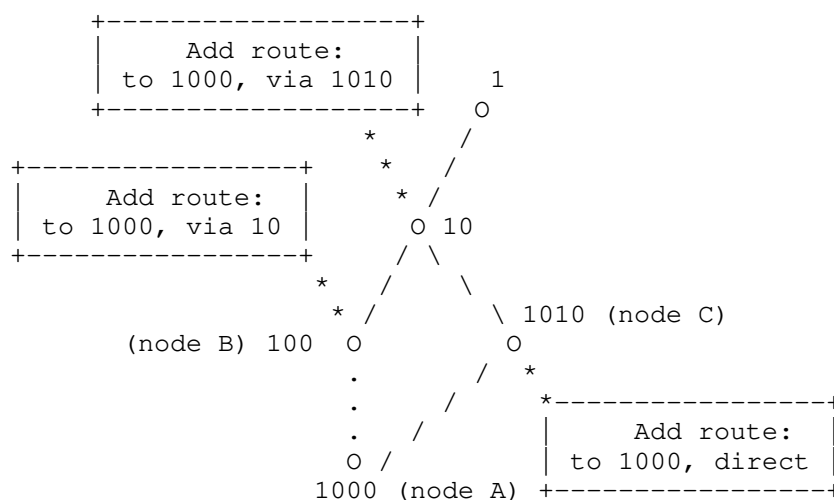


Figure 6: Add extra routes for "logic topology change"

## 7. NSA Header Format

As explained in Section 4, the addresses in NSA are of variable length, in this section, we outline the design of the header format partially based on the format of 6lowpan, accommodating the variable length property in the packet. The header format is shown in Figure 7.

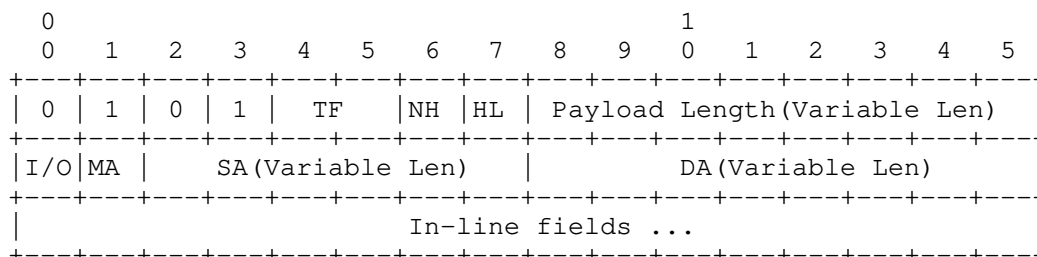


Figure 7: Header format of NSA packets

The first 4 bits are new dispatch types that will be introduced in Section 9.

- \* TF: The same definition as in [RFC6282] Section 3.1.1.
- \* NH: The same definition as in [RFC6282] Section 3.1.1.
- \* HL: This field indicates the hop limit. When HL is 0, a hop limit field defined in [RFC2460] locates in in-line fields, while HL is 1 means no hop limit header in packet.
- \* Payload length is a variable length field. It normally occupies an octet assuming most packets are smaller than 252 bytes. For larger packets, payload length may expand to 2 to 3 octets. The encoding method is defined as follows. When the first octet has value of:
  - 0~252: Indicates how many octets the payload consist of.
  - 253: Indicates that there is an extra octet for payload length, with the actual length value equal to the last byte value plus 252.
  - 254: Indicates that there is an extra two octets for payload length, with the actual length value equal to the value of the second byte multiple 256 plus value of the last byte plus 252.
  - 255: Reserved.
- \* I/O: Indicates whether this packet is destined to a inner-domain node (value '1') or an outer-domain node (value '0'), where the former means from an NSA or IPv6 node to a NSA destination, while the latter means to an external IPv6 node.

- \* MA: Indicates the source address is actually a Mapped Address generated by the root. When it is '1', the source address of the packet is a mapped address of an external IPv6 address, while if it is '0', the source address of the packet is an NSA address.

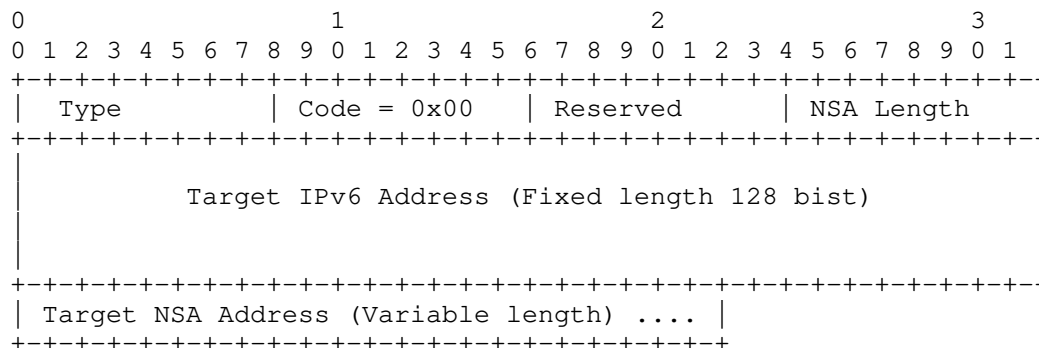
For length variable native short address encoding, for both Source Address (SA) and Destination Address (DA), the definition is:

- \* 0~252: if the address value locates in this interval, one octet is used to encode the value
- \* 253: indicates that the address is encoded in 2 octets.
- \* 254: indicates that the following 4 octets encode the address.
- \* 255: indicates that the following octet defines the length of address in octets, followed by the address value octets.

The sequence of in-line fields is as per [RFC8200] section 3.

## 8. NSA Control Message

This documents specifies only one NSA Control Message, namely the NSA Mapped Address Advertisement described in Section 4. The purpose of such a message is advertise the mapping of an IPv6 address into a NSA address. The map is performed by the root node and sent to the node originating the communication. The root keeps a copy of the mapping to be used for future packets. The format is as follows:



- \* Type: Type value identifying NSA Control Message. Value to be assigned by IANA (cf. Section 9)
- \* Code: This field identifies the specific control message. In this case it is set to the value 0x00 "NSA Mapped Address for External IPv6 Address".



- \* Reserved: Set as 0 on transmission and ignore on reception.
- \* NSA Length: This field indicates the length of the Target NSA Address at the end of the message, expressed in octets.

The "NSA Mapped Address for External IPv6 Address" is a variable length message, however, the first five fields of the message, namely Type, Codem Reserved, NSA Length, and Target IPv6 address, have a fixed length of 160 bits (20 octets), hence the length of the NSA address is sufficient to calculate the length of the entire packet: 20 octets + "NSA length".

## 9. IANA Considerations

### 9.1. Dispatch Type Field

This document requires IANA to assign the range 01010000 to 01011111 in page 10 of the "Dispatch Type Field" registry as follows:

Bit Pattern	Page	Header Type	Reference
0101TTNH	10	LOWPAN NSA IP (LOWPAN_NIP)	[This Document]

Figure 8: LOWPAN Dispatch Type Field requested allocation

### 9.2. Allocation Function Registry

This section provides guidance to the Internet Assigned Numbers Authority (IANA) regarding registration of values related to the NSA specification, in accordance with BCP 26 [RFC8126].

IANA is asked to create a registry named "Native Short Addresses (NSA) Parameters".

Such registry should be populated with a one octet sub registry named "Allocation Function" and used to identify the AF used in a NSA deployment. The sub registry is populated as follows:

Value	AF Name	Reference
0x00	Native Allocation Function	[This Document]
0x01-0xFF	Un-assigned	

Values can be assigned by IANA on a "First Come, First Served" basis according to [RFC8126].

### 9.3. ICMP NSA Control Message

IANA is requested to allocate an ICMPv6 type value from the "ICMPv6 Parameters" registry to be used by "NSA Control Message".

Also IANA is requested to create an "NSA Control Codes" sub registry, for the Code field of the ICMPv6 NSA Control Message.

New codes may be allocated through the "Specification Required" procedure as defined in [RFC8126]. The following code is currently defined (the others are to be marked as un-assigned):

Code	Description	Reference
0x00	NSA Mapped Address for External IPv6 Address	[This Document]

## 10. Security Considerations

An extended security analysis will be provided in future revision of this document. As of this point we consider that the security considerations of [RFC4944], [RFC6282] apply.

## 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>>.
- [RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, DOI 10.17487/RFC2460, December 1998, <<https://www.rfc-editor.org/info/rfc2460>>.
- [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>>.

- [RFC6282] Hui, J., Ed. and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks", RFC 6282, DOI 10.17487/RFC6282, September 2011, <<https://www.rfc-editor.org/info/rfc6282>>.
- [RFC6550] Winter, T., Ed., Thubert, P., Ed., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, JP., and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", RFC 6550, DOI 10.17487/RFC6550, March 2012, <<https://www.rfc-editor.org/info/rfc6550>>.
- [RFC8025] Thubert, P., Ed. and R. Cragie, "IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Paging Dispatch", RFC 8025, DOI 10.17487/RFC8025, November 2016, <<https://www.rfc-editor.org/info/rfc8025>>.
- [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <<https://www.rfc-editor.org/info/rfc8126>>.
- [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>>.
- [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>>.

## 11.2. Informative References

- [IoTSurvey] Oliveira, A. and T. Vazão, "Low-power and lossy networks under mobility: A survey", Computer Networks Vol. 107, pp. 339–352, DOI 10.1016/j.comnet.2016.03.018, October 2016, <<https://doi.org/10.1016/j.comnet.2016.03.018>>.
- [LEE10] Lee, M., Zhang, R., Zheng, J., Ahn, G., Zhu, C., Park, T., Cho, S., Shin, C., and J. Ryu, "IEEE 802.15.5 WPAN mesh standard-low rate part: Meshing the wireless sensor networks", IEEE Journal on Selected Areas in Communications Vol. 28, pp. 973–983, DOI 10.1109/jsac.2010.100902, September 2010, <<https://doi.org/10.1109/jsac.2010.100902>>.

- [LPWAN] "IPv6 over Low Power Wide-Area Networks (lpwan) WG", n.d., <<https://datatracker.ietf.org/wg/lpwan/about/>>.
- [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>>.
- [RFC5673] Pister, K., Ed., Thubert, P., Ed., Dwars, S., and T. Phinney, "Industrial Routing Requirements in Low-Power and Lossy Networks", RFC 5673, DOI 10.17487/RFC5673, October 2009, <<https://www.rfc-editor.org/info/rfc5673>>.
- [RFC8138] Thubert, P., Ed., Bormann, C., Toutain, L., and R. Cragie, "IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Routing Header", RFC 8138, DOI 10.17487/RFC8138, April 2017, <<https://www.rfc-editor.org/info/rfc8138>>.
- [RFC8724] Minaburo, A., Toutain, L., Gomez, C., Barthel, D., and JC. Zúñiga, "SCHC: Generic Framework for Static Context Header Compression and Fragmentation", RFC 8724, DOI 10.17487/RFC8724, April 2020, <<https://www.rfc-editor.org/info/rfc8724>>.
- [RFC8799] Carpenter, B. and B. Liu, "Limited Domains and Internet Protocols", RFC 8799, DOI 10.17487/RFC8799, July 2020, <<https://www.rfc-editor.org/info/rfc8799>>.
- [SIXLO] "IPv6 over Networks of Resource-constrained Nodes (6lo) WG", n.d., <<https://datatracker.ietf.org/wg/6lo/about/>>.
- [SIXLOWPAN] "IPv6 over Low power WPAN (6lowpan) - Concluded WG", n.d., <<https://datatracker.ietf.org/wg/6lowpan/about/>>.
- [ZigBee] "ZigBee Wireless Networks and Transceivers", Elsevier book, DOI 10.1016/b978-0-7506-8393-7.x0001-5, 2008, <<https://doi.org/10.1016/b978-0-7506-8393-7.x0001-5>>.

## Authors' Addresses

Guangpeng Li  
Huawei Technologies  
Beiqing Road, Haidian District  
Beijing  
100095  
China

Email: [liguangpeng@huawei.com](mailto:liguangpeng@huawei.com)

David Lou  
Huawei Technologies Duesseldorf GmbH  
Riesstrasse 25  
80992 Munich  
Germany

Email: [zhe.lou@huawei.com](mailto:zhe.lou@huawei.com)

Luigi Iannone  
Huawei Technologies France S.A.S.U.  
18, Quai du Point du Jour  
92100 Boulogne-Billancourt  
France

Email: [luigi.iannone@huawei.com](mailto:luigi.iannone@huawei.com)

Peng Liu  
China Mobile  
No. 53, Xibianmen Inner Street, Xicheng District  
Beijing  
100053  
China

Email: [liupengyjy@chinamobile.com](mailto:liupengyjy@chinamobile.com)

Network Working Group  
Internet-Draft  
Intended status: Experimental  
Expires: 14 April 2022

H. Song  
Futurewei Technologies  
11 October 2021

Short Hierarchical IP Addresses at Edge Networks  
draft-song-ship-edge-02

Abstract

To mitigate the IPv6 header overhead in edge networks, this draft proposes to use short hierarchical addresses excluding the network prefix within edge networks. An edge network can be further organized into a hierarchical architecture containing one or more levels of networks. The border routers for each hierarchical level are responsible for address augmenting and pruning when a packet leaves or enter a lower level network. Specifically, the top-level border routers convert the internal IP header to and from the standard IPv6 header. This draft presents an incrementally deployable scheme allowing packet header to be effectively compressed in edge networks without affecting the network interoperability.

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.

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 14 April 2022.

## Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	2
2. Short Hierarchical Address in Edge Networks . . . . .	3
2.1. Edge Network Hierarchy . . . . .	3
2.2. Address Fields . . . . .	5
2.3. Router Roles and Function . . . . .	6
3. Deployment and Interoperability Consideration . . . . .	9
3.1. Control Plane . . . . .	9
3.2. Data Plane . . . . .	11
3.3. Using NAT for the edge network . . . . .	11
3.4. Extension Beyond IPv6 . . . . .	12
4. Security Considerations . . . . .	12
5. IANA Considerations . . . . .	12
6. Acknowledgments . . . . .	12
7. References . . . . .	12
7.1. Normative References . . . . .	12
7.2. Informative References . . . . .	12
Author's Address . . . . .	13

## 1. Introduction

Internet of Things (IoT) and 5G introduce to the Internet a huge number of addressable entities (e.g., sensors, machines, vehicles, and robots). The transition to IPv6 is inevitable. While the 128-bit address of IPv6 was considered large enough and future-proof, the long IP addresses inflate the packet header size. 80% of a basic IPv6 header is consumed by addresses.

In IoT networks, thing-to-thing communication through wireless connections is dominant, which presents several distinct characteristics. (1) The communication pattern is often frequent short-message exchanges (e.g., industry robots and networked vehicles). (2) The communication is usually energy sensitive (e.g.,

battery-powered sensors). (3) The communication often requires low latency (e.g., industry control). (4) The precious wireless channels demand high bandwidth utilization (e.g., ZigBee, Bluetooth, Wi-Fi, and 5G). These characteristics render a large header overhead unfavorable and even prohibitive.

The address overhead also takes its toll on Data Center Networks (DCN), especially when large scale containers are deployed, the east-west traffic is dominant, and the prevailing communications are comprised of short messages (e.g., key-value pairs) and conducted through virtual switches.

In IoT and DCN, since most communications happen between adjacent and related entities, it is a good practice to locally confine communication, computing, and storage due to performance, efficiency, and security considerations, as advocated by Edge Computing. Such a communication pattern provides an opportunity to mitigate the IPv6 header overhead problem due to the long addresses.

When an IPv6 address block is allocated to an edge network, all the entities in the edge network share the same address prefix. When these entities communicate with each other, they can ignore the common prefix. In fact, they do not even need to know the common prefix. Only when they need to communicate with entities outside of the edge network, the full addresses are needed. Even in this case, the entities in the edge network still do not need to know the prefix. It is sufficient for the gateway routers at the network border to manipulate the addresses (i.e., augmenting or pruning the address) to meet the addressing requirement.

Following this line of thought, an edge network can be further partitioned into multiple hierarchical levels, which support flexible sub-networking. The result is that an end entity needs to maintain an even shorter address as its identifier. For communication crossing network levels, the address manipulation is done at each gateway router on the path recursively.

## 2. Short Hierarchical Address in Edge Networks

### 2.1. Edge Network Hierarchy

In this draft, we define an edge network as a stub network which does not support traffic transit service. The stub network is assigned an IPv6 address block. In this sense, a data center network in cloud can also be considered as an edge network. An edge network usually falls under a single network administration domain.



The address block assigned to an edge network is identified by a prefix  $P$  with the length of  $L < 128$  bits. The remaining  $S = 128 - L$  bits can be used to assign addresses to the entities in this network. A key observation is: the entities in this network do not need to be aware of  $P$ 's length and value at all. We can further partition the edge network into multiple hierarchical levels, making a tree architecture. The root represents the entire edge network. Each other node represents a lower level network occupying a sub address space owned by its parent node. A leaf node represents a lowest level network. We name the root level network the  $L_0$  network. Its children are all  $L_1$  networks, and so on so forth. In other words, the network level is the depth of the corresponding node in the tree.

The network hierarchy partitions the  $S$ -bit address into multiple sections. Assume an entity is in an  $L_n$  network. The  $S$ -bit address is partitioned into  $n+1$  sections. The entity only needs to keep the last section of the  $S$ -bit address as its ID. The gateway routers for each level of network maintain one section of the  $S$ -bit address. Specifically, the gateway routers of  $L_i$  ( $i > 0$ ) keep the  $i$ -th section of the  $S$ -bit address, and the gateway routers of  $L_0$  keep the assigned IPv6 address block prefix  $P$ .

Figure 1 shows an edge network example, in which are three network levels. The edge network A is assigned a 96-bit IPv6 address prefix (2001:0db8:ac10:fe01::0001), which means it owns a 32-bit address space. In this space, two  $L_1$  networks are created: B with a 16-bit prefix (0xaaaa) and C with a 24-bit prefix (0xcccccc). Note that the prefixes at the same level must not overlap in order to guarantee entities in the edge network are uniquely addressable. Network B contains two entities  $x$  and  $y$ , and Network C contains one entity  $z$ . In network B, an  $L_2$  network D is further created with a 8-bit prefix (0xbb). In this example, an entity in C or D (e.g.,  $m$  and  $z$ ) only need to own a 8-bit address, an entity in B but not in D (e.g.,  $x$  and  $y$ ) needs to own a 16-bit address, and an entity in A but not in B and C needs to own a 32-bit address. In this way, each entity in A still logically owns a unique IPv6 address (e.g., the IPv6 address of the entity  $m$  in D with ID of 5 is 2001:0db8:ac10:fe01::0001:aaaa:bb05), although the entity  $m$  is only aware of its local ID (0x05).

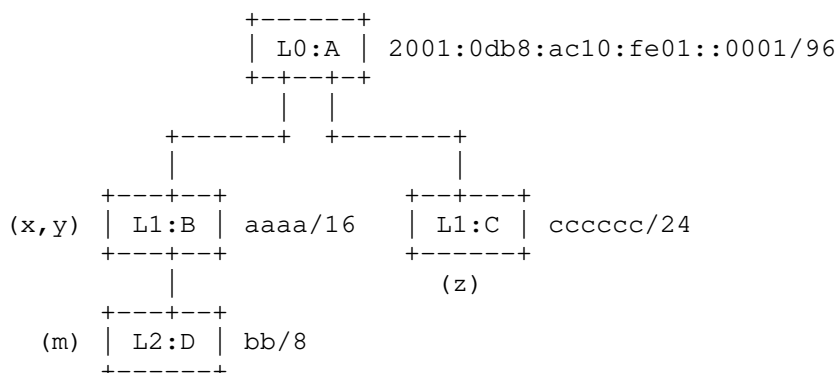


Figure 1: A Hierarchical Edge Network Example

## 2.2. Address Fields

The edge networks adopting the short and variable size address scheme need a new type of IP header, which is referred as IPvn in this draft. Apart from the IP version, the major difference between IPvn and IPv6 headers is the address fields. IPvn replaces IPv6's 128-bit source address field and 128-bit destination address field with the four fields shown in Figure 2.

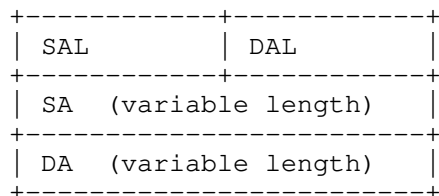


Figure 2: IPvn Address Fields

The Source Address Length (SAL) and the Destination Address Length (DAL) fields have fixed length, while the Source Address (SA) and the Destination Address (DA) fields are of variable length. To simplify the implementation, SA and DA are preferred to be byte-aligned. It is possible to define the length of address in the unit of byte, nibble, or bit. Each has its own pros and cons. The unit of byte can help reduce the size of the SAL/DAL but results in coarse network granularity which might be inefficient in address allocation. For example, a 3-bit SAL/DAL is enough to encode 8 possible address lengths (one to eight bytes) for networks. In this design, each higher level network's address space expands 256 times. On the other

extreme, the unit of bit allows fine network granularity but requires more space for SAL/DAL. For example, 6-bit SAL and DAL can support an address length up to 64 bits (8 bytes) and each higher level network is only twice larger.

With a few bits, it is also possible to design a more sophisticated encoding scheme that supports variable address length steps and adapts to the ideal network sizes at different levels.

Assuming SA and DA are 2 bytes each, and SAL and DAL are 4 bits each, the address fields are only 5 bytes in total. Comparing to IPv6, the size of the address fields is reduced by 84%.

### 2.3. Router Roles and Function

In the edge network hierarchy, each network has one or more Level Gateway Routers (LGR) which are responsible for forwarding packets in or out of this network. The LGRs are the only interface between a network and its parent network.

A network can be in a single L2 domain, which means all the entities in this network (excluding those in its child networks) and all the network devices (including the LGRs to the parent network and the child networks) are L2 reachable. A network can also be a pure L3 network in which no L2 device is allowed. Each entity in a network is directly connected to either an LGR or some internal routers named Intra-Level Router (ILR) which is solely responsible for packet forwarding within the network. In this case, the entities need to partially participate in the routing process (e.g., advertising its address).

The scale of an intra-level network can be used to guide the L2/L3 selection. Small networks prefer the L2-based solution and large networks prefer the L3-based solution. In the higher level networks (e.g., closer to the top level network or the tree root), since the number of entities is usually small, it is free to choose between L2 or L3-based solution. The leaf level networks are usually L2-based for simplicity.

Unlike in IPv4 and IPv6 networks, the address related fields in IPvn header can be modified by LGRs. An LGR of a network keeps a prefix that can augment the SAs from this network to an address outside of this network. If an LGR needs to forward an internal packet outside (i.e.,  $DAL > SAL$ ), it augments the packet's SA and updates its SAL accordingly. Reversely, if an LGR receives a packet from the parent network destined for the child network for which it serves as a gateway (i.e., the parent network prefix matches the DA's prefix), it strips off the parent network prefix from the packet's DA and updates its DAL accordingly.

In contrast, within an L3-based level network, ILRs do not modify the address fields. An ILR can decide the packet forwarding direction by examining the DAL. If  $DAL > SAL$ , the packet needs to be forwarded to an LGR of this network; otherwise, the packet needs to be forwarded within the current network, and possibly into a lower-level child network.

An LGR of the top-level network (i.e., the L0 network) is special. In addition to the address manipulation, it is also responsible for converting the IPvn header to and from the standard IPv6 header to support the Internet interoperability. We name such a router IP Translator (IPT).

We use the edge network shown in Figure 1 to illustrate some packet forwarding examples. The details for the involved entities are summarized in Figure 3. In the IPvn packet header, we use 4 bits to encode the address length. In particular, 0b0000 is used to indicate the address is 16 bytes long (i.e., a complete IPv6 address).

Entity	ID	Length	Level	Network	Prefix
x	0x0001	2bytes	1	B	0xaaaa/16
y	0x0002				
z	0x01	1byte	1	C	0xcccccc/24
m	0x08	1byte	2	D	0xbb/8

Figure 3: Entity Address Configuration

The first example in Figure 4 shows how packets are forwarded from x to y within the same network B. In this case, the source address and destination address have the same length. The packets only pass through an ILR which does not change the address fields.

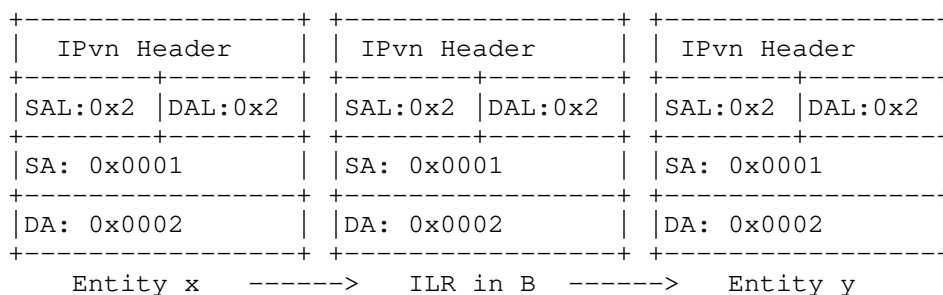


Figure 4: Forward within a network level in the edge

The second example in Figure 5 shows how packets are forwarded from x in B to z in C. At LGR of B, the source address is augmented, and at the LGR of C, the destination address is pruned. Since x and z's nearest common ancestor network is A, so the packets never need to leave network A, so A's prefix is oblivious throughout the communication.

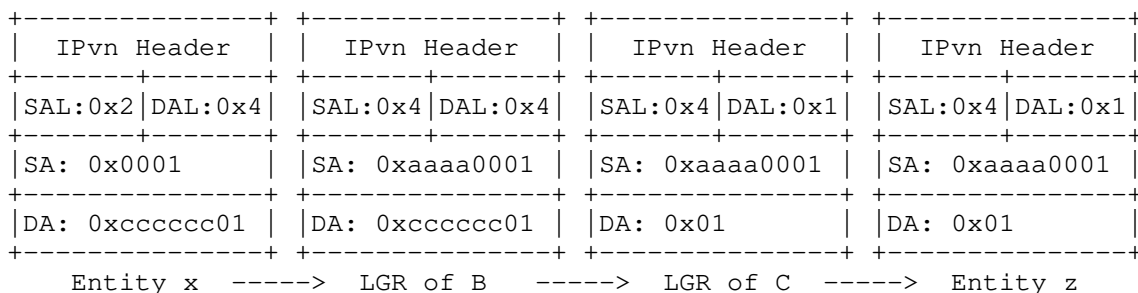


Figure 5: Forward to another network in the edge

The last example in Figure 6 shows how packets are forwarded from x in B to a host in IPv6 domain. In the IPT of A, the IPvn header is converted to an IPv6 header.

IPvn Header	IPvn Header	IPv6 Header	IPv6 Header
SAL:0x2   DAL:0x0	SAL:0x4   DAL:0x0	SA: 2001:0db8 ac10:fe01: 0000:0001: aaaa:0001	SA: 2001:0db8 ac10:fe01: 0000:0001: aaaa:0001
SA: 0x0001	SA: 0xaaaa0001		
DA: 2001:0db8: 85a3:0000: 0000:8a2e: 0370:7334	DA: 2001:0db8: 85a3:0000: 0000:8a2e: 0370:7334	DA: 2001:0db8: 85a3:0000: 0000:8a2e: 0370:7334	DA: 2001:0db8: 85a3:0000: 0000:8a2e: 0370:7334

Entity x -----> LGR of B -----> IPT of A -----> Entity n

Figure 6: Forward out of the edge network

### 3. Deployment and Interoperability Consideration

#### 3.1. Control Plane

Within the edge networks where IPvn is applied, all the control plane functions and protocols need to be modified or redesigned due to the hierarchical network architecture of IPvn. Fortunately, the updates are often incremental and the results are usually simpler than their counterparts in IPv4 and IPv6. We briefly discuss a few essential protocols that enable the operation of IPvn.

**DHCP:** An entity can be manually configured or dynamically acquire its address when booting up. Each network in the edge network hierarchy may contain a Dynamic Host Configuration Protocol (DHCP) server responsible for assigning addresses (i.e., IDs) to the entities in the same network. The protocol is almost identical to that for IPv4 and IPv6, except that the assigned address length is adaptive to the allocated network size.

**DNS:** For an entity to acquire the address of a peer entity in order to initiate a communication, Domain Name System (DNS) is the prominent approach but with a new service model. Any network in the hierarchy can provide name service. Each entity is configured with the address of the closest DNS server on the path to the root network. The hierarchical network architecture allows a scoped domain name service. That is, a name registered in a network is only valid in this network and its child networks. It is possible that a same name is registered in two networks and one network is the other's ancestor. Such name conflict is not a bug but a

feature for name reuse, which is transparent to the name query process. In this case, the name resolved from the closer DNS server is used.

Each network may contain a DNS server (the notation is only logical. The actual implementation may follow the same hierarchical and distributed architecture of today's DNS). Each DNS server knows the nearest DNS server in a higher level network and the nearest DNS servers in lower level networks. This essentially organizes the DNS servers in the same tree structure as the hierarchical network. Each named entity in a network is registered with the DNS server that covers its scope, which is basically a subtree.

We have several methods to populate the name to support the scoped name queries, each with different storage and performance trade-off: 1) register the name in all the DNS servers in its scope (i.e., all the subtree nodes); 2) recursively register the name in every parent DNS server until the scope root; and 3) register the name only in the DNS server in its scope root. The address for a name returned by a DNS server is on a "need-to-know" basis. In a network, if the address's prefix matches the query's address prefix, the prefix is removed. This can be easily done by the original or the relay DNS servers. If a query cannot be resolved by the DNS server in the L0 network, the query, after the IP protocol translation is done, exits the IPvn domain and enters into the IPv4/IPv6 domain to a public DNS server. When the response comes back and enters the edge network, the result can be cached by the DNS servers on the path.

ARP: In a L2-based network, the operation of Address Resolution Protocol (ARP) or Neighbor Discovery Protocol (NDP) is almost identical to that for IPv4 and IPv6. In an L2- based network, each immediate entity should be configured with a default gateway address to its parent network. If no default gateway is configured, a network LGR should be configured as an ARP proxy to respond to all internal ARP requests for addresses out of the network. Similarly, the LGRs to any child network of this network are also needed to be configured as ARP proxy to response all ARP requests for addresses in that network. Due to the multi-homing gateway routers, an ARP request may receive multiple responses. It is up to the requester to determine which one to cache.

Routing Protocol: The entire edge network may belong to a single AS, so the interior gateway routing protocols (IGP) such as OSPF and IS-IS can be used. Other child networks in this network can be considered OSPF stub areas or IS-IS levels. A simpler way is that each network run an independent instance of OSPF or IS-IS.

Specially, an LGR at a network border runs two OSPF/IS-IS instances: one for the upper-level network and the other for the lower-level network. The hierarchical architecture solves the routing protocol scalability issue, and simplifies the protocol implementation by removing unnecessary features. The clean routing scope helps to secure the infrastructure and troubleshoot the networks.

### 3.2. Data Plane

**IPvn Socket for End Entities:** To enable IPvn as a new network layer protocol in end entities, we need to add the protocol implementation in the OS Kernel and allow applications to invoke the socket API using the address family parameter AF\_INETN. The L4-L7 protocol stack and the application logic remains the same, allowing direct communication between entities in IPvn domain and in IPv4/IPv6 domain.

**Forwarding Table Lookups in Networks:** The short hierarchical address simplifies the router forwarding table structure in L3-based networks. A forwarding table only contains the addresses to local entities and the prefixes to the child networks. Since there is no nested prefixes, the Longest Prefix Matching (LPM) is not necessary. The small number of unique prefix lengths allows the prefixes to be grouped on lengths and each group to be implemented as a hash table. A lookup can search all the hash tables in parallel, and at most one table can result a positive match. This design avoids the use of expensive TCAM or other complex trie-based algorithms.

An LGR between an  $L_i$  network and an  $L_{(i+1)}$  network has two types of interfaces: one faces the  $L_i$  network and the other faces the  $L_{(i+1)}$  network. One LGR may serve more than one  $L_{(i+1)}$  network. Hence, an LGR may contain multiple logical forwarding tables, with each for a network. For a packet in LGR, once its target network is determined and the address related fields are processed, the proper forwarding table can be searched.

### 3.3. Using NAT for the edge network

To expand the address space of the edge network, the IPT of the edge network can also support functions similar to NAT. In this case, the edge network is assigned one or more public IPv4/IPv6 addresses. The entities in IPvn domain use private addresses. The IPT maintains the mapping table between the private address and public address.



### 3.4. Extension Beyond IPv6

Although the motivation of this draft is to support shorter address (i.e., smaller L3 header overhead) in edge networks, it is worth noting that the scheme allows the addresses to be extended to arbitrary length, even longer than 128bits. In that case, the address space of the IPv6 network can be greater than that of IPv4 and the entire IPv6 network can be considered an edge network of the IPv6 network. This scenario should be considered when specifying the address fields of IPv6.

## 4. Security Considerations

The addressing scheme and architecture allow a securer edge network. The IPTs and LGRs naturally support the access control.

## 5. IANA Considerations

The proposal requires to use a new IP version and define a new IP header which can be converted to/from an equivalent IPv6 header.

## 6. Acknowledgments

We acknowledge the technical contributions, suggestions and comments from Yingzhe Qu, Zhaobo Zhang, James Guichard, Toerless Eckert, Stewart Bryant, and Michael McBride.

## 7. References

### 7.1. 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>>.
- [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>>.

### 7.2. Informative References

- [RFC7775] Ginsberg, L., Litkowski, S., and S. Previdi, "IS-IS Route Preference for Extended IP and IPv6 Reachability", RFC 7775, DOI 10.17487/RFC7775, February 2016, <<https://www.rfc-editor.org/info/rfc7775>>.

Author's Address

Haoyu Song  
Futurewei Technologies  
Santa Clara,  
United States of America

Email: [haoyu.song@futurewei.com](mailto:haoyu.song@futurewei.com)

6lo  
Internet-Draft  
Intended status: Standards Track  
Expires: 31 March 2022

P. Thubert, Ed.  
E.L.A. Levy-Abegnoli  
Cisco Systems  
27 September 2021

IPv6 Neighbor Discovery Unicast Lookup  
draft-thubert-6lo-unicast-lookup-01

Abstract

This document updates RFC 8505 in order to enable unicast address lookup from a 6LoWPAN Border Router acting as an Address Registrar.

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 31 March 2022.

Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	2
2. Terminology . . . . .	3
2.1. BCP 14 . . . . .	3
2.2. References . . . . .	3
2.3. New Terms . . . . .	4
2.4. Acronym Definitions . . . . .	4
3. Overview . . . . .	5
4. Updating RFC 8505 . . . . .	7
4.1. Extended Neighbor Discovery Options and Messages . . . . .	7
4.1.1. Extending the Capability Indication Option . . . . .	7
4.1.2. New Code Prefix for Address Mapping Messages . . . . .	8
4.1.3. New ARO Status . . . . .	8
4.2. Address Mapping Messages . . . . .	9
4.3. IPv6 ND-based Address Lookup . . . . .	10
5. Backward Compatibility . . . . .	10
6. Security Considerations . . . . .	10
7. IANA Considerations . . . . .	10
7.1. ICMP Codes . . . . .	11
7.2. New ARO Status values . . . . .	12
7.3. New 6LoWPAN Capability Bits . . . . .	12
8. Acknowledgments . . . . .	12
9. Normative References . . . . .	12
10. Informative References . . . . .	13
Authors' Addresses . . . . .	14

## 1. Introduction

[RFC8505] defines the Routing Registrar and extends [RFC6775] to use a 6LoWPAN Border Router (6LBR) as a central service for Address Registration and duplicate detection amongst Routing Registrars and possibly individual Nodes that access it directly.

[RFC8929] introduces the Backbone Router (6BBR) as a Routing Registrar that performs IPv6 ND [RFC4861] [RFC4862] proxy operation between IPv6 Nodes on a federating Backbone Link and Registering Nodes attached to a LowPower Lossy Networks (LLNs) that register their addresses to the 6BBR. The federated links form a Multilink Subnet (MLSN).

The 6BBRs may exchange Extended Duplicate Address Messages (EDAR and EDAC) [RFC8505] to register the proxied addresses on behalf of the Registering Nodes to the 6LBR. The Registration Ownership Verifier (ROVR) field in the EDAR and EDAC messages is used to correlate attempts to register the same address and to detect duplications. The ROVR can also be used as a proof-of-ownership (see [RFC8928]) to protect the Registered address against theft and impersonation

attacks (more in [I-D.bi-savi-wlan]). Conflicting registrations to different 6BBRs for the same Registered address are resolved using the TID field, which creates a temporal order and enables to recognize the freshest registration.

With [RFC8929], the Link Layer address (LLA) that the 6BBR advertises for a Registered address on behalf of the Registered Node over the Backbone can belong to the Registering Node; in that case, the 6BBR acts as a Bridging Proxy and bridges the unicast packets. Alternatively, the LLA can be that of the 6BBR on the Backbone interface, in which case the 6BBR acts as a Routing Proxy, that receives the unicast packets at Layer-3 and routes them. The 6BBR signals that LLA in a Source LLA Option (SLLAO) in the EDAR messages to the 6LBR, and the 6LBR responds with a Target LLA Option (TLLAO) that indicates the LLA associated to the current registration.

It results that the 6LBR is capable of providing the LLA mapping for any address that was proactively registered with an SLLAO. This draft defines the protocol elements and the operations to try a unicast lookup with the 6LBR. This may save a reactive IPv6 ND Neighbor Solicitation (NS) message, which is based on multicast and may be problematic in extensive wireless domains (see [I-D.ietf-mboned-ieee802-mcast-problems]) as well as in large switched fabrics.

The registration and lookup services that the 6LBR provides do not have to be limited to 6BBRs and are available to any node that supports [RFC8505] and [RFC8929] to register an address, and / or this specification to resolve a mapping. The services are available on-link using an IPv6 NDP NS and off-link using a new variation of the Extended Duplicate Address messages called Address Mapping Messages. The policy and security settings that allow the access to the 6LBR are out of scope.

## 2. Terminology

### 2.1. BCP 14

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.

### 2.2. References

This document uses terms and concepts that are discussed in:

- \* "Neighbor Discovery for IP version 6" [RFC4861] and "IPv6 Stateless address Autoconfiguration" [RFC4862],
- \* Neighbor Discovery Optimization for Low-Power and Lossy Networks [RFC6775], as well as
- \* "Registration Extensions for 6LoWPAN Neighbor Discovery" [RFC8505] and "IPv6 Backbone Router" [RFC8929].

### 2.3. New Terms

This document introduces the following terminology:

**Address Mapping Request** An ICMP message with an ICMP type of 157 (DAR) and a Code Prefix of 1.

**Address Mapping Confirm** An ICMP message with an ICMP type of 158 (DAC) and a Code Prefix of 1.

This document uses terminology defined in [RFC8505], in particular:

**Address Registrar** The Address Registrar is an abstract database that is maintained by the 6LBR to store the state associated with its registrations.

**Address Registration** An Address Registration is an abstract state associated to one registration, in other words one entry in the Address Registrar.

### 2.4. Acronym Definitions

This document uses the following acronyms:

6BBR 6LoWPAN Backbone Router

6BBR 6LoWPAN Border Router

6LR 6LoWPAN Router

6CIO Capability Indication Option

AMC Address Mapping Confirmation

AMR Address Mapping Request

ARO Address Registration Option

DAC Duplicate Address Confirmation

DAD Duplicate Address Detection  
DAR Duplicate Address Request  
EDAC Extended Duplicate Address Confirmation  
EDAR Extended Duplicate Address Request  
DODAG Destination-Oriented Directed Acyclic Graph  
LLN Low-Power and Lossy Network  
NA Neighbor Advertisement  
NCE Neighbor Cache Entry  
ND Neighbor Discovery  
NS Neighbor Solicitation  
ROVR Registration Ownership Verifier  
RA Router Advertisement  
RS Router Solicitation  
TID Transaction ID

### 3. Overview

Figure 1 illustrates a Backbone Link that federates a collection of LLNs as a single IPv6 Subnet, with a number of 6BBRs providing proxy-ND services to their attached LLNs.

A collection of IPv6 Nodes are present on the Backbone and use IPv6 ND [RFC4861][RFC4862] procedures for DAD and Lookup.

The LLN may be a hub-and-spoke access link such as (Low-Power) IEEE STD. 802.11 (Wi-Fi) [IEEEstd80211] and IEEE STD. 802.15.1 (Bluetooth) [IEEEstd802151], or a Mesh-Under or a Route-Over network [RFC8505].

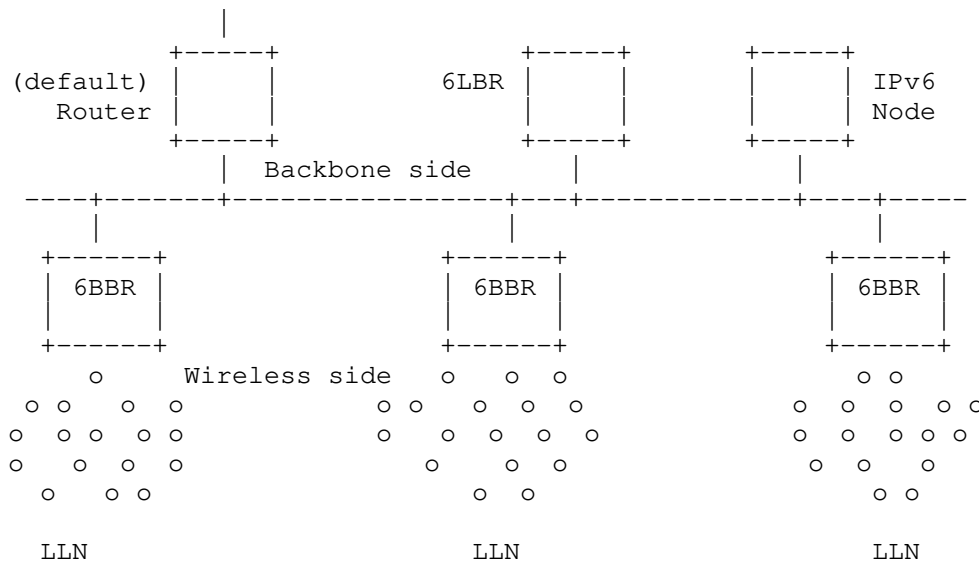


Figure 1: Backbone Link and 6LBR

A 6LBR provides registration services for the purpose of proactive IPv6 ND and maintains a registry of the active registrations as an abstract data structure called an Address Registrar. An entry in the Address Registrar is called an "Address Registration".

The Address Registration retains:

- \* the value for the ROVR associated to the registration, the current value of the TID, and the remaining Lifetime.
- \* a list of LLAs that are associated with the IPv6 address and can be used in a TLLAO as a response to a lookup.

Examples where more than one address may be available include the case of an anycast address and the case of an LLN address that is proxied by more than one 6BBR.

Unless otherwise configured, a 6LBR does the following:

- \* The 6LBR maintains an entry in the Address Registrar for any type of unicast and anycast addresses including those with link-local scope.
- \* Based on that entry, it provides duplicate avoidance services within the scope of its Address Registrar.



- \* The 6LBR also provides address lookup services for the Registered Address using unicast ICMPv6 DAR and DAC-based Address Mapping messages.

The Address Mapping messages can be exchanged using global unicast addresses as source and destination addresses, so they can be used for both on-link and off-link queries. NS and NA messages may also be used, but in that case the unicast source and destination addresses are link-local addresses and the 6LBR must be on-link.

The 6LBR proactive operations may coexist on the Backbone with reactive IPv6 ND [RFC4861][RFC4862] that rely on multicast for Duplicate Address Detection (DAD) and Address Lookup. Nodes that support this specification operate with the 6LBR before attempting the reactive operation, which may be avoided if the 6LBR is conclusive, either detecting a duplication or returning a mapping.

#### 4. Updating RFC 8505

This specification leverages the capability to insert IPv6 ND options in the EDAR and EDAC messages that was introduced in [RFC8929].

It extends DAR and DAR ICMP messages for address lookup in Section 4.1.2 that use the same ICMP types as EDAR and EDAC but a different Code Prefix.

It also adds a new Status "Not Found" in Section 4.1.3) that indicates that the address being searched is not present in the Address Registrar.

A 6LBR signals itself by setting the "B" bit in the 6CIO of the RA messages that it generates [RFC8505]. This specification adds a new "A" bit in the 6CIO to indicate support of address mapping (see Section 4.1.1).

#### 4.1. Extended Neighbor Discovery Options and Messages

This specification does not introduce new options; it modifies existing options and updates the associated behaviors.

##### 4.1.1. Extending the Capability Indication Option

This specification defines a new capability bit for use in the 6CIO, as defined by [RFC7400] and extended in [RFC8505] for use in IPv6 ND messages.

The new "A" bit indicates that the 6LBR provides address mapping services per this specification.

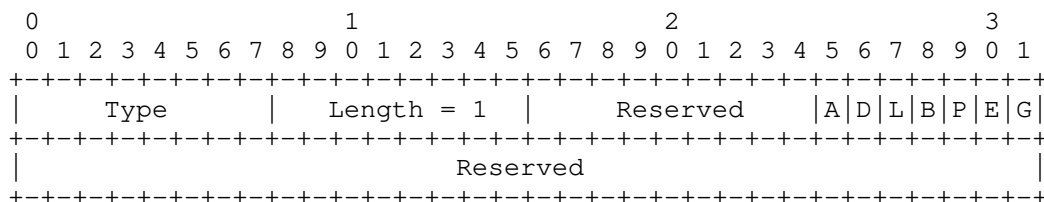


Figure 2: New Capability Bits in the 6CIO

Option Fields:

Type 36

A The 6LBR provides address mapping services.

#### 4.1.2. New Code Prefix for Address Mapping Messages

The Extended Duplicate Address messages share a common base format defined in section 4.2 of [RFC8505], with the ICMP type respectively set to 157 and 158 that is inherited from the DAR and DAC messages defined in section 4.4 of [RFC6775]. The ICMP Code is split in two 4-bit fields, the Code Prefix and the Code Suffix, and the only Code Prefix defined in [RFC8505] is 0, signaling a DAD.

The Address Mapping messages use the same values for the ICMP Type as the corresponding Extended Duplicate Address messages. This specification adds the Code Prefix of 1 to signal Address Mapping. ICMP messages with the ICMP type set to 157 or 158, and a Code Prefix of 1 are thus respectively an Address Mapping Request (AMR) and an Address Mapping Confirm (AMC).

#### 4.1.3. New ARO Status

The Extended Address Registration Option (EARO) is defined in section 4.1 of [RFC8505]. It contains a Status field that is common with the EDAR and EDAC messages defined in section 4.2 of [RFC8505]. This specification defines a new Status "Not Found" as indicated in Table 1

Value	Description
0..10	As defined in [RFC6775] and [RFC8505]
11	Not Found: The address is not present in the Address Registrar (value to be confirmed by IANA)

Table 1: EARO Status

The Status of "Not Found" can be used in an NA(EARO) and in an AMC messages as a response to an address lookup operation.

#### 4.2. Address Mapping Messages

A 6LBR signals that support by setting the "B" bit in the 6CIO of the RA messages that it generates. A 6LBR that supports this specification MUST also set the "A" bit, indicating support of the Address Mapping messages for address lookup.

In the Address Mapping flow, the querier IPv6 Node uses an AMR message, which is characterized by an ICMPv6 Type of 157 and a Code Prefix of 1. When used on-link, the AMR message SHOULD carry a SLLAO indicating the LLA of the querier. The Code Suffix MUST be set to 0 indicating a ROVR Length of 64 bits. The ROVR, TID and Lifetime fields MUST be set to 0 and ignored by the receiver.

The 6LBR MUST respond with an AMC message, which is characterized by an ICMPv6 Type of 158 and a Code Prefix of 1.

- \* If the address is not present in the Address Registrar then the 6LBR MUST set the status to "Not Found". The Code Suffix MUST be set to 0 indicating a ROVR Length of 64 bits. The ROVR, TID and Lifetime fields MUST be set to 0 and ignored by the receiver.
- \* Else if the address is present in the Address Registrar then the AMC fields MUST be set from the ROVR, TID and remaining Lifetime values in the Address Registration and the Status MUST be set to 0.
- \* If at least one LLA is found in the Address Registration, then the 6LBR MUST place one in a TLLAO option in the AMC message.

The AMC is sent unicast the 6LBR to the querier.

#### 4.3. IPv6 ND-based Address Lookup

A 6LBR that is deployed on-link SHOULD provide NS/NA-based services. It signals that support by setting the "L" bit in the 6CIO of the RA messages that it generates, indicating that it is a 6LR [RFC8505].

A 6LBR thus typically sets the "A", the "B", and the "L" bits when attached to a Backbone Link that it serves, as illustrated in Figure 1. In that case, the IPv6 Nodes and 6BBRs can use an NS/NA exchange with the 6LBR for both duplicate detection and lookup services.

The NS(Lookup) is sent unicast from link-local address of the querier to the link-local address of the 6LBR. It carries a SLLAO [RFC4861] and it MUST NOT carry an EARO option to avoid the confusion with a registration.

The 6LBR MUST respond with an NA message that contains an EARO.

- \* If the address is not present in the Address Registrar then the 6LBR MUST set the status to "Not Found". The ROVR, TID and Lifetime fields MUST be set to 0 and ignored by the receiver.
- \* Else if the address is present in the Address Registrar then the EARO fields MUST be set from the ROVR, TID and remaining Lifetime values in the Address Registration and the Status MUST be set to 0.
- \* If at least one LLA is found in the Address Registration, then the 6LBR MUST place one in a TLLAO option in the NA message.

The NA is sent unicast from link-local address of the 6LBR to the link-local address of the querier.

#### 5. Backward Compatibility

#### 6. Security Considerations

This specification extends [RFC8505], and the security section of that document also applies to this document. In particular, the link layer SHOULD be sufficiently protected to prevent rogue access.

#### 7. IANA Considerations

Note to RFC Editor, to be removed: please replace "This RFC" throughout this document by the RFC number for this specification once it is allocated.

IANA is requested to make a number of changes under the "Internet Control Message Protocol version 6 (ICMPv6) Parameters" registry, as follows.

### 7.1. ICMP Codes

IANA is requested to create 2 new subregistries of the ICMPv6 "Code" Fields registry, which itself is a subregistry of the Internet Control Message Protocol version 6 (ICMPv6) Parameters for the ICMP codes.

The new subregistries relate to the ICMP type 157, Duplicate Address Request (shown in Table 2), and 158, Duplicate Address Confirmation (shown in Table 3), respectively. For those two ICMP types, the ICMP Code field is split into 2 subfields, the "Code Prefix" and the "Code Prefix". The new subregistries relate to the "Code Prefix" portion of the ICMP Code. The range of "Code Prefix" is 0..15 in all cases. The policy is "IETF Review" or "IESG Approval" [RFC8126] for both subregistries.

The new subregistries are to be initialized as follows:

Code Prefix	Meaning	Reference
0	Duplicate Address Detection	Duplicate Address Detection
1	Address Mapping	This RFC
2...15	Duplicate Address Detection	

Table 2: New Code Prefixes for ICMP type 157 DAR message

Code Prefix	Meaning	Reference
0	Duplicate Address Detection	Duplicate Address Detection
1	Address Mapping	This RFC
2...15	Duplicate Address Detection	

Table 3: New Code Prefixes for ICMP type 158 DAC message

## 7.2. New ARO Status values

IANA is requested to make additions to the Address Registration Option Status Values Registry as follows:

ARO Status	Meaning	Reference
11 (suggested)	Not Found	This RFC

Table 4: New ARO Status values

## 7.3. New 6LoWPAN Capability Bits

IANA is requested to make additions to the Subregistry for "6LoWPAN Capability Bits" as follows:

Capability Bit	Meaning	Reference
9 (suggested)	AM Support (A bit)	This RFC

Table 5: New 6LoWPAN Capability Bits

## 8. Acknowledgments

## 9. 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>>.
- [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>>.

- [RFC6775] Shelby, Z., Ed., Chakrabarti, S., Nordmark, E., and C. Bormann, "Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", RFC 6775, DOI 10.17487/RFC6775, November 2012, <<https://www.rfc-editor.org/info/rfc6775>>.
- [RFC7400] Bormann, C., "6LoWPAN-GHC: Generic Header Compression for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", RFC 7400, DOI 10.17487/RFC7400, November 2014, <<https://www.rfc-editor.org/info/rfc7400>>.
- [RFC8126] Cotton, M., Leiba, B., and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 8126, DOI 10.17487/RFC8126, June 2017, <<https://www.rfc-editor.org/info/rfc8126>>.
- [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>>.
- [RFC8505] Thubert, P., Ed., Nordmark, E., Chakrabarti, S., and C. Perkins, "Registration Extensions for IPv6 over Low-Power Wireless Personal Area Network (6LoWPAN) Neighbor Discovery", RFC 8505, DOI 10.17487/RFC8505, November 2018, <<https://www.rfc-editor.org/info/rfc8505>>.

#### 10. Informative References

- [RFC8928] Thubert, P., Ed., Sarikaya, B., Sethi, M., and R. Struik, "Address-Protected Neighbor Discovery for Low-Power and Lossy Networks", RFC 8928, DOI 10.17487/RFC8928, November 2020, <<https://www.rfc-editor.org/info/rfc8928>>.
- [RFC8929] Thubert, P., Ed., Perkins, C.E., and E. Levy-Abegnoli, "IPv6 Backbone Router", RFC 8929, DOI 10.17487/RFC8929, November 2020, <<https://www.rfc-editor.org/info/rfc8929>>.
- [I-D.ietf-mboned-ieee802-mcast-problems]  
Perkins, C. E., McBride, M., Stanley, D., Kumari, W., and J. C. Zuniga, "Multicast Considerations over IEEE 802 Wireless Media", Work in Progress, Internet-Draft, draft-ietf-mboned-ieee802-mcast-problems-15, 28 July 2021, <<https://datatracker.ietf.org/doc/html/draft-ietf-mboned-ieee802-mcast-problems-15>>.
- [I-D.bi-savi-wlan]  
Bi, J., Wu, J., Lin, T., and Y. Wang, "A SAVI Solution for WLAN", Work in Progress, Internet-Draft, draft-bi-savi-

wlan-21, 10 May 2021,  
<<https://datatracker.ietf.org/doc/html/draft-bi-savi-wlan-21>>.

[I-D.thubert-6man-ipv6-over-wireless]

Thubert, P., "IPv6 Neighbor Discovery on Wireless Networks", Work in Progress, Internet-Draft, draft-thubert-6man-ipv6-over-wireless-09, 17 May 2021,  
<<https://datatracker.ietf.org/doc/html/draft-thubert-6man-ipv6-over-wireless-09>>.

[IEEEstd80211]

IEEE standard for Information Technology, "IEEE Standard for Information technology -- Telecommunications and information exchange between systems Local and metropolitan area networks-- Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications".

[IEEEstd802151]

IEEE standard for Information Technology, "IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements. - Part 15.1: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Wireless Personal Area Networks (WPANs)".

#### Authors' Addresses

Pascal Thubert (editor)  
Cisco Systems, Inc  
Building D  
45 Allée des Ormes - BP1200  
06254 Mougins - Sophia Antipolis  
France

Phone: +33 497 23 26 34  
Email: [pthubert@cisco.com](mailto:pthubert@cisco.com)

Eric Levy-Abegnoli  
Cisco Systems, Inc  
Building D  
45 Allée des Ormes - BP1200  
06254 MOUGINS - Sophia Antipolis  
France



Phone: +33 497 23 26 20

Email: [elevyabe@cisco.com](mailto:elevyabe@cisco.com)