6Lo Working Group
Internet-Draft
Intended status: Informational
Expires: February 27, 2015

Y. Hong
Y. Choi
ETRI
J. Youn
DONG-EUI Univ
D. Kim
KNU
JH. Choi
Samsung Electronics Co.,
August 26, 2014

Transmission of IPv6 Packets over Near Field Communication
draft-hong-6lo-ipv6-over-nfc-02

Abstract

Near field communication (NFC) is a set of standards for smartphones and portable devices to establish radio communication with each other by touching them together or bringing them into proximity, usually no more than 10 cm. NFC standards cover communications protocols and data exchange formats, and are based on existing radio-frequency identification (RFID) standards including ISO/IEC 14443 and FeliCa. The standards include ISO/IEC 18092 and those defined by the NFC Forum. The NFC technology has been widely implemented and available in mobile phones, laptop computers, and many other devices. This document describes how IPv6 is transmitted over NFC using 6LowPAN techiques.

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 http://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 February 27, 2015.

Table of Contents

1. Introduction

NFC is a set of short-range wireless technologies, typically requiring a distance of 10 cm or less. NFC operates at 13.56 MHz on ISO/IEC 18000-3 air interface and at rates ranging from 106 kbit/s to

424 kbit/s. NFC always involves an initiator and a target; the initiator actively generates an RF field that can power a passive target. This enables NFC targets to take very simple form factors such as tags, stickers, key fobs, or cards that do not require batteries. NFC peer-to-peer communication is possible, provided both devices are powered. NFC builds upon RFID systems by allowing two-way communication between endpoints, where earlier systems such as contactless smart cards were one-way only. It has been used in devices such as mobile phones, running Android operating system, named with a feature called "Android Beam". In addition, it is expected for the other mobile phones, running the other operating systems (e.g., iOS, etc.) to be equipped with NFC technology in the near future.

Considering the potential for exponential growth in the number of heterogeneous air interface technologies, NFC would be widely used as one of the other air interface technologies, such as Bluetooth Low Energy (BT-LE), Wi-Fi, and so on. Each of the heterogeneous air interface technologies has its own characteristics, which cannot be covered by the other technologies, so various kinds of air interface technologies would be existing together. Therefore, it is required for them to communicate each other. NFC also has the strongest point (e.g., secure communication distance of 10 cm) to prevent the third party from attacking privacy.

When the number of devices and things having different air interface technologies communicate each other, IPv6 is an ideal internet protocols owing to its large address space. Also, NFC would be one of the endpoints using IPv6. Therefore, This document describes how IPv6 is transmitted over NFC using 6LoWPAN techiques with following scopes.

- Overview of NFC technologies;
- Specifications for IPv6 over NFC;
  - Neighbor Discovery;
  - Addressing and Configuration;
  - Header Compression;
  - Fragmentation & Reassembly for a IPv6 datagram;

RFC4944 [1] specifies the transmission of IPv6 over IEEE 802.15.4. The NFC link also has similar characteristics to that of IEEE 802.15.4. Many of the mechanisms defined in the RFC4944 [1] can be

applied to the transmission of IPv6 on NFC links. This document specifies the details of IPv6 transmission over NFC links.

2. Conventions and Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [2].

3. Overview of Near Field Communication Technology

NFC technology enables simple and safe two-way interactions between electronic devices, allowing consumers to perform contactless transactions, access digital content, and connect electronic devices with a single touch. NFC complements many popular consumer level wireless technologies, by utilizing the key elements in existing standards for contactless card technology (ISO/IEC 14443 A&B and JIS-X 6319-4). NFC can be compatible with existing contactless card infrastructure and it enables a consumer to utilize one device across different systems.

Extending the capability of contactless card technology, NFC also enables devices to share information at a distance that is less than 10 cm with a maximum communication speed of 424 kbps. Users can share business cards, make transactions, access information from a smart poster or provide credentials for access control systems with a simple touch.

NFC’s bidirectional communication ability is ideal for establishing connections with other technologies by the simplicity of touch. In addition to the easy connection and quick transactions, simple data sharing is also available.

3.1. Peer-to-peer Mode for IPv6 over NFC

NFC-enabled devices are unique in that they can support three modes of operation: card emulation, peer-to-peer, and reader/writer. Peer-to-peer mode enables two NFC-enabled devices to communicate with each other to exchange information and share files, so that users of NFC-enabled devices can quickly share contact information and other files with a touch. Therefore, a NFC-enabled device can securely send IPv6 packets to any corresponding node on the Internet when a NFC-enabled gateway is linked to the Internet.

3.2. Protocol Stacks in IPv6 over NFC

The IP protocol can use the services provided by Logical Link Control Protocol (LLCP) in the NFC stack to provide reliable, two-way transport of information between the peer devices. Figure 1 depicts the NFC P2P protocol stack with IPv6 bindings to the LLCP.

For data communication in IPv6 over NFC, an IPv6 packet SHALL be received at LLCP of NFC and transported to an Information Field in Protocol Data Unit (I PDU) of LLCP of the NFC-enabled peer device. Since LLCP does not support fragmentation and reassembly, Upper Layers SHOULD support fragmentation and reassembly. For IPv6 addressing or address configuration, LLCP SHALL provide related information, such as link layer addresses, to its upper layer. LLCP to IPv6 protocol Binding SHALL transfer the SSAP and DSAP value to the IPv6 over NFC protocol. SSAP stands for Source Service Access Point, which is 6-bit value meaning a kind of Logical Link Control (LLC) address, while DSAP means a LLC address of destination NFC-enabled device.

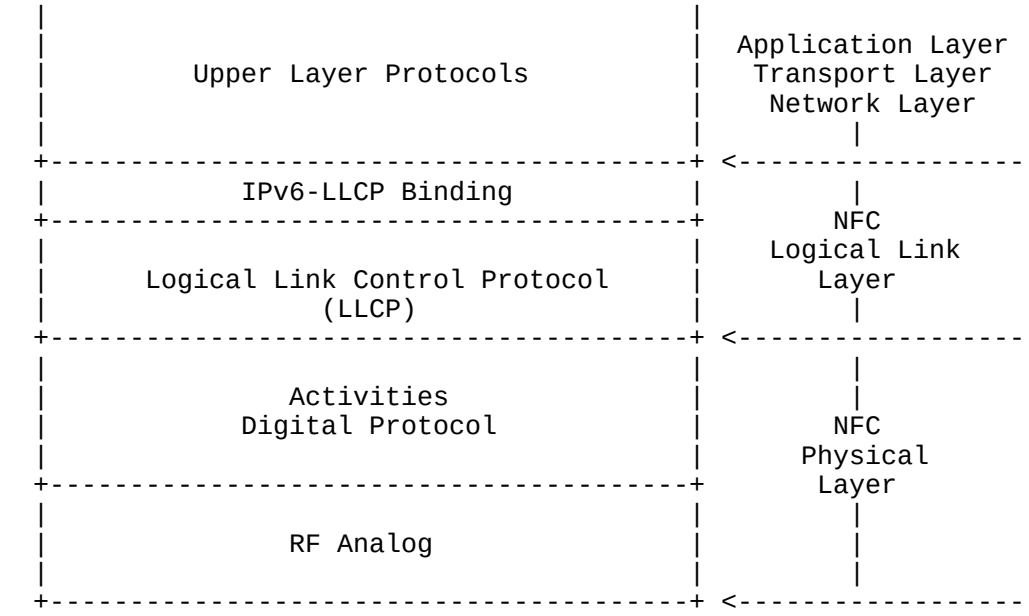


Figure 1: Protocol Stack of NFC

3.3. NFC-enabled Device Addressing

NFC-enabled devices are identified by 6-bit LLC address. In other words, Any address SHALL be usable as both an SSAP and a DSAP address. According to NFCForum-TS-LLCP_1.1 [3], address values between 0 and 31 (00h - 1Fh) SHALL be reserved for well-known service access points for Service Discovery Protocol (SDP). Address values between 32 and 63 (20h - 3Fh) inclusively, SHALL be assigned by the local LLC as the result of an upper layer service request.

3.4. NFC Packet Size and MTU

As mentioned in Section 3.2, an IPv6 packet SHALL be received at LLCP of NFC and transported to an Information Field in Protocol Data Unit (I PDU) of LLCP of the NFC-enabled peer device. The format of the I PDU SHALL be as shown in Figure 2.

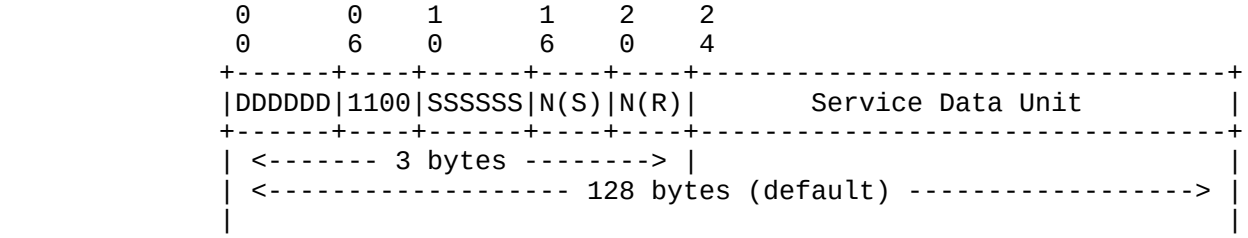


Figure 2: Format of the I PDU in NFC

The I PDU sequence field SHALL contain two sequence numbers: The send sequence number N(S) and the receive sequence number N(R). The send sequence number N(S) SHALL indicate the sequence number associated with this I PDU. The receive sequence number N(R) value SHALL indicate that I PDUs numbered up through N(R) - 1 have been received correctly by the sender of this I PDU and successfully passed to the senders SAP identified in the SSAP field. These I PDUs SHALL be considered as acknowledged.

The information field of an I PDU SHALL contain a single service data unit. The maximum number of octets in the information field SHALL be determined by the Maximum Information Unit (MIU) for the data link connection. The default value of the MIU for I PDUs SHALL be 128 octets. The local and remote LLCs each establish and maintain distinct MIU values for each data link connection endpoint. Also, An LLC MAY announce a larger MIU for a data link connection by transmitting an MIUX extension parameter within the information field.

4. Specification of IPv6 over NFC

NFC technology sets also has considerations and requirements owing to low power consumption and allowed protocol overhead. 6LoWPAN standards RFC4944 [1], RFC6775 [4], and RFC6282 [5] provide useful functionality for reducing overhead which can be applied to BT-LE. This functionality comprises of link-local IPv6 addresses and stateless IPv6 address auto-configuration (see Section 4.3), Neighbor Discovery (see Section 4.4) and header compression (see Section 4.5).

One of the differences between IEEE 802.15.4 and NFC is that the former supports both star and mesh topology (and requires a routing protocol), whereas NFC can support direct peer-to-peer connection and simple mesh-like topology depending on NFC application scenarios because of very short RF distance of 10 cm or less.

4.1. Protocol Stack

Figure 3 illustrates IPv6 over NFC. Upper layer protocols can be transport protocols (TCP and UDP), application layer, and the others capable running on the top of IPv6.

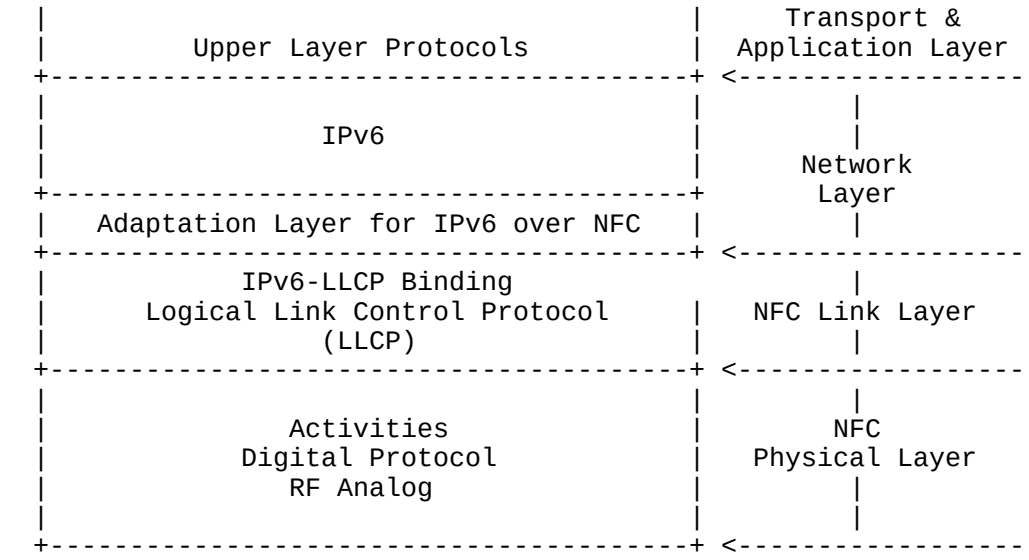


Figure 3: Protocol Stack for IPv6 over NFC

Adaptation layer for IPv6 over NFC SHALL support neighbor discovery, address auto-configuration, header compression, and fragmentation & reassembly.

4.2. Link Model

In the case of BT-LE, Logical Link Control and Adaptation Protocol (L2CAP) supports fragmentation and reassembly (FAR) functionality; therefore, adaptation layer for IPv6 over BT-LE do not have to conduct the FAR procedure. However, NFC link layer is similar to IEEE 802.15.4. Adaptation layer for IPv6 over NFC SHOULD support FAR functionality. Therefore, fragmentation functionality as defined in RFC4944 [1] SHALL be used in NFC-enabled device networks.

The NFC link between two communicating devices is considered to be a point-to-point link only. Unlike in BT-LE, NFC link does not consider star topology and mesh network topology but peer-to-peer topology and simple multi-hop topology. Due to this characteristics, 6LoWPAN functionality, such as addressing and auto-configuration, and header compression, is specialized into NFC.

4.3. Stateless Address Autoconfiguration

A NFC-enabled device (i.e., 6LN) performs stateless address autoconfiguration as per RFC4862 [6]. A 64-bit Interface identifier (IID) for a NFC interface MAY be formed by utilizing the 6-bit NFC LLCP address (i.e., SSAP or DSAP) (see Section 3.3). In the case of NFC-enabled device address, the "Universal/Local" bit MUST be set to 0 RFC4291 [7]. Only if the NFC-enabled device address is known to be a public address the "Universal/Local" bit can be set to 1. As defined in RFC4291, the IPv6 link-local address for a NFC-enabled device is formed by appending the IID, to the prefix FE80::/64, as depicted in Figure 4.

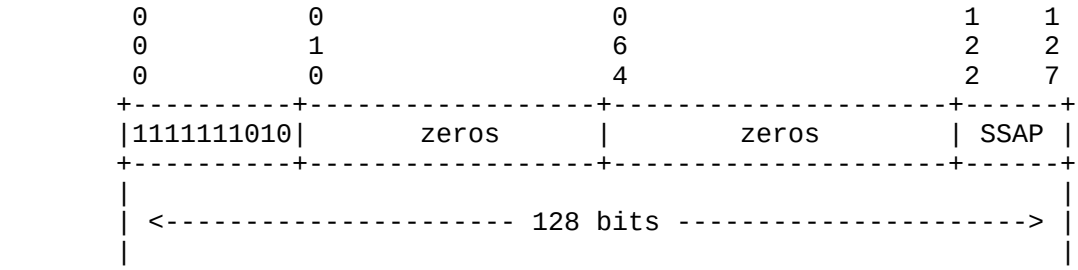


Figure 4: IPv6 link-local address in NFC

The tool for a 6LBR to obtain an IPv6 prefix for numbering the NFC network is can be accomplished via DHCPv6 Prefix Delegation (RFC3633 [8]).

4.4. Neighbor Discovery

Neighbor Discovery Optimization for 6LoWPANs (RFC6775 [4]) describes the neighbor discovery approach in several 6LoWPAN topologies, such as mesh topology. NFC does not consider complicated mesh topology but simple multi-hop network topology or directly connected peer-to-peer network. Therefore, the following aspects of RFC6775 are applicable to NFC:

1. In a case that a NFC-enabled device (6LN) is directly connected to 6LBR, A NFC 6LN MUST register its address with the 6LBR by sending a Neighbor Solicitation (NS) message with the Address Registration Option (ARO) and process the Neighbor Advertisement (NA) accordingly. In addition, DHCPv6 is used to assigned an address, Duplicate Address Detection (DAD) is not required.

2. For sending Router Solicitations and processing Router Advertisements the NFC 6LNs MUST follow Sections 5.3 and 5.4 of the RFC6775.

4.5. Header Compression

Header compression as defined in RFC6282 [5] , which specifies the compression format for IPv6 datagrams on top of IEEE 802.15.4, is REQUIRED in this document as the basis for IPv6 header compression on top of NFC. All headers MUST be compressed according to RFC6282 encoding formats.

If a 16-bit address is required as a short address of IEEE 802.15.4, it MUST be formed by padding the 6-bit NFC link-layer (node) address to the left with zeros as shown in Figure 5.

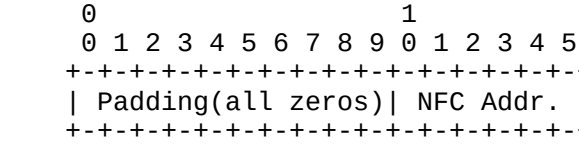


Figure 5: NFC short adress format

4.6. Fragmentation and Reassembly

Fragmentation and reassembly (FAR) as defined in RFC4944, which specifies the fragmentation methods for IPv6 datagrams on top of IEEE 802.15.4, is REQUIRED in this document as the basis for IPv6 datagram FAR on top of NFC. All headers MUST be compressed according to RFC4944 encoding formats, but the default MTU of NFC is 128 bytes. This MUST be considered.

4.7. Unicast Address Mapping

The address resolution procedure for mapping IPv6 non-multicast addresses into NFC link-layer addresses follows the general description in Section 7.2 of RFC4861 [9], unless otherwise specified.

The Source/Target link-layer Address option has the following form when the addresses are 6-bit NFC link-layer (node) addresses.

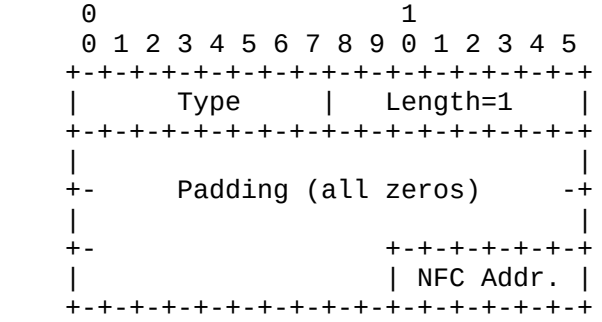


Figure 6: Unicast address mapping

Option fields:

Type:

1: for Source Link-layer address.

2: for Target Link-layer address.

Length:

This is the length of this option (including the type and length fields) in units of 8 octets. The value of this field is 1 for 6-bit NFC node addresses.

NFC address:

The 6-bit address in canonical bit order. This is the unicast address the interface currently responds to.

4.8. Multicast Address Mapping

All IPv6 multicast packets MUST be sent to NFC Destination Address 255 (broadcast) and filtered at the IPv6 layer. When represented as

a 16-bit address in a compressed header, it MUST be formed by padding on the left with a zero.

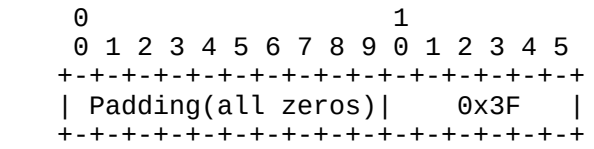


Figure 7: Multicast address mapping

5. Internet Connectivity Scenarios

As two typical scenarios, the NFC network can be isolated and connected to the Internet.

5.1. NFC-enabled Device Connected to the Internet

One of the key applications by using adaptation technology of IPv6 over NFC is the most securely transmitting IPv6 packets because RF distance between 6LN and 6LBR SHOULD be within 10 cm. If any third party wants to hack into the RF between them, it MUST come to nearly touch them. Applications can choose which kinds of air interfaces (e.g., BT-LE, Wi-Fi, NFC, etc.) to send data depending characteristics of data. NFC SHALL be the best solution for secured and private information.

Figure 8 illustrates an example of NFC-enabled device network connected to the Internet. Distance between 6LN and 6LBR SHOULD be 10 cm or less. If there is any of close laptop computers to a user, it SHALL becomes the 6LBR. Additionally, When the user mounts a NFC-enabled air interface adapter (e.g., portable small NFC dongle) on the close laptop PC, the user’s NFC-enabled device (6LN) can communicate the laptop PC (6LBR) within 10 cm distance.

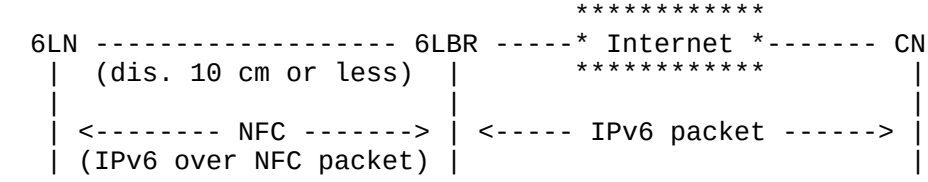


Figure 8: NFC-enabled device network connected to the Internet

5.2. Isolated NFC-enabled Device Network

In some scenarios, the NFC-enabled device network may transiently be a simple isolated network as shown in the Figure 9.

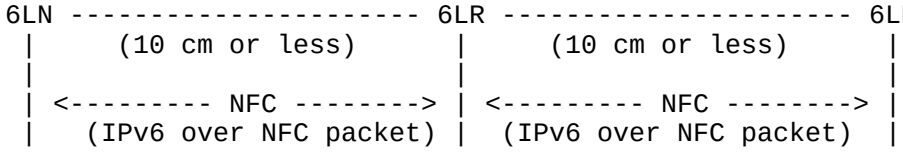


Figure 9: Isolated NFC-enabled device network

In mobile phone markets, applications are designed and made by user developers. They may image interesting applications, where three or more mobile phones touch or attach each other to accomplish outstanding performance. For instance, three or more mobile phones can play multi-channel sound of music together. In addition, attached three or more mobile phones can make an extended banner to show longer sentences in a concert hall.

6. IANA Considerations

There are no IANA considerations related to this document.

7. Security Considerations

The method of deriving Interface Identifiers from 6-bit NFC Link layer addresses is intended to preserve global uniqueness when it is possible. Therefore, it is to required to protect from duplication through accident or forgery.

8. References

8.1. Normative References

[1] Montenegro, G., Kushalnagar, N., Hui, J., and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks", RFC 4944, September 2007.

[2] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[3] "Logical Link Control Protocol version 1.1", NFC Forum Technical Specification , June 2011.

[4] Shelby, Z., Chakrabarti, S., Nordmark, E., and C. Bormann, "Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", RFC 6775, November 2012.

[5] Hui, J. and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks", RFC 6282, September 2011.

[6] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.

[7] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

[8] Troan, O. and R. Droms, "IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6", RFC 3633, December 2003.

[9] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.

8.2. Informative References

[10] "Near Field Communication - Interface and Protocol (NFCIP-1) 3rd Ed.", ECMA-340 , June 2013.

Authors' Addresses

Yong-Geun Hong
ETRI
161 Gajeong-Dong Yuseung-Gu
Daejeon 305-700
Korea

Phone: +82 42 860 6557
Email: yghong@etri.re.kr

Younghwan Choi
ETRI
218 Gajeongno, Yuseong
Daejeon 305-700
Korea

Phone: +82 42 860 1429
Email: yhc@etri.re.kr

Joo-Sang Youn
DONG-EUI University
176 Eomgwangno Busan_jin_gu
Busan 614-714
Korea

Phone: +82 51 890 1993
Email: joosang.youn@gmail.com

Dongkyun Kim
Kyungpook National University
80 Daehak-ro, Buk-gu
Daegu 702-701
Korea

Phone: +82 53 950 7571
Email: dongkyun@knu.ac.kr

JinHyouk Choi
Samsung Electronics Co.,
129 Samsung-ro, Youngdong-gu
Suwon 447-712
Korea

Phone: +82 2 2254 0114
Email: jinchoe@samsung.com

IPv6 maintenance Working Group (6man)
Internet-Draft
Updates: 2464, 2467, 2470, 2491, 2492, 2497, 2590, 3146, 3572, 4291, 4338, 4391, 5072, 5121 (if approved)
Intended status: Standards Track
Expires: April 11, 2015

F. Gont
SI6 Networks / UTN-FRH
A. Cooper
Cisco
D. Thaler
Microsoft
W. Liu
Huawei Technologies

October 8, 2014

Recommendation on Stable IPv6 Interface Identifiers
draft-ietf-6man-default-iids-01

Abstract

The IPv6 addressing architecture defines Modified EUI-64 format Interface Identifiers, and the existing IPv6 over various link-layers specify how such identifiers are derived from the underlying link-layer address (e.g., an IEEE LAN MAC address) when employing IPv6 Stateless Address Autoconfiguration (SLAAC). The security and privacy implications of embedding hardware addresses in the Interface Identifier have been known and understood for some time now, and some popular IPv6 implementations have already deviated from such schemes to mitigate these issues. This document changes the recommended default Interface Identifier generation scheme to that specified in RFC7217, and recommends against embedding hardware addresses in IPv6 Interface Identifiers. It formally updates RFC2464, RFC2467, RFC2470, RFC2491, RFC2492, RFC2497, RFC2590, RFC3146, RFC3572, RFC4291, RFC4338, RFC4391, RFC5072, and RFC5121, wihch require IPv6 Interface Identifiers to be derived from the underlying link-layer address.

This Internet-Draft will expire on April 11, 2015.

Table of Contents

1. Introduction

[RFC4862] specifies Stateless Address Autoconfiguration (SLAAC) for IPv6 [RFC2460], which typically results in hosts configuring one or more "stable" addresses composed of a network prefix advertised by a local router, and an Interface Identifier (IID) [RFC4291] that typically embeds a hardware address (e.g., an IEEE LAN MAC address).

The security and privacy implications of embedding a hardware address in an IPv6 Interface ID have been known for some time now, and are discussed in great detail in [I-D.ietf-6man-ipv6-address-generation-privacy]; they include:

- Network activity correlation
- Location tracking

- o Address scanning
- o Device-specific vulnerability exploitation

Some popular IPv6 implementations have already deviated from the traditional stable IID generation scheme to mitigate the aforementioned security and privacy implications [Microsoft].

As a result of the aforementioned issues, this document recommends the implementation of an alternative scheme ([RFC7217]) as the default stable Interface-ID generation scheme, such that the aforementioned issues are mitigated.

NOTE: [RFC4291] defines the "Modified EUI-64 format" for Interface identifiers. Appendix A of [RFC4291] then describes how to transform an IEEE EUI-64 identifier, or an IEEE 802 48-bit MAC address from which an EUI-64 identifier is derived, into an interface identifier in the Modified EUI-64 format.

2. Terminology

Stable address:
An address that does not vary over time within the same network (as defined in [I-D.ietf-6man-ipv6-address-generation-privacy].

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

3. Generation of IPv6 Interface Identifiers

Nodes SHOULD NOT employ IPv6 address generation schemes that embed the underlying hardware address in the Interface Identifier. Namely, nodes SHOULD NOT generate Interface Identifiers with the schemes specified in [RFC2464], [RFC2467], [RFC2470], [RFC2491], [RFC2492], [RFC2497], [RFC2590], [RFC3146], [RFC3572], [RFC4338], [RFC4391], [RFC5121], and [RFC5072].

Nodes SHOULD implement and employ [RFC7217] as the default scheme for generating stable IPv6 addresses with SLAAC.

Future specifications SHOULD NOT specify IPv6 address generation schemes that embed the underlying hardware address in the Interface Identifier.

4. Future Work

At the time of this writing, the mechanisms specified in the following documents are not compatible with the recommendations in this document:

- RFC 6282 [RFC6282]
- RFC 4944 [RFC4944]
- RFC 6755 [RFC6775]

It is expected that that future revisions or updates of these documents will address the aforementioned issues such that the requirements in this documents can be enforced.

5. IANA Considerations

There are no IANA registries within this document. The RFC-Editor can remove this section before publication of this document as an RFC.

6. Security Considerations

This document recommends [RFC7217] as the default scheme for generating IPv6 stable addresses with SLAAC, such that the security and privacy issues of Interface IDs that embed hardware addresses are mitigated.

7. Acknowledgements

The authors would like to thank Erik Nordmark and Ray Hunter for providing a detailed review of this document.

The authors would like to thank (in alphabetical order) Fred Baker, Scott Brim, Brian Carpenter, Samita Chakrabarti, Tim Chown, Lorenzo Colitti, Jean-Michel Combes, Greg Daley, Esko Dijk, Ralph Droms, David Farmer, Brian Haberman, Ulrich Herberg, Bob Hinden, Jahangir Hossain, Jonathan Hui, Ray Hunter, Sheng Jiang, Roger Jorgensen, Dan Luedtke, George Mitchel, Erik Nordmark, Simon Perreault, Tom Petch, Alexandru Petrescu, Michael Richardson, Arturo Servin, Mark Smith, Tom Taylor, Ole Troan, Tina Tsou, and Randy Turner, for providing valuable comments on earlier versions of this document.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC2464] Crawford, M., "Transmission of IPv6 Packets over Ethernet Networks", RFC 2464, December 1998.

[RFC2467] Crawford, M., "Transmission of IPv6 Packets over FDDI Networks", RFC 2467, December 1998.

[RFC2470] Crawford, M., Narten, T., and S. Thomas, "Transmission of IPv6 Packets over Token Ring Networks", RFC 2470, December 1998.

[RFC2492] Armitage, G., Schulter, P., and M. Jork, "IPv6 over ATM Networks", RFC 2492, January 1999.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.

[RFC7217] Gont, F., "A Method for Generating Semantically Opaque Interface Identifiers with IPv6 Stateless Address Autoconfiguration (SLAAC)", RFC 7217, April 2014.

[RFC2491] Armitage, G., Schulter, P., Jork, M., and G. Harter, "IPv6 over Non-Broadcast Multiple Access (NBMA) networks", RFC 2491, January 1999.

[RFC2497] Souvatzis, I., "Transmission of IPv6 Packets over ARCnet Networks", RFC 2497, January 1999.

[RFC2590] Conta, A., Malis, A., and M. Mueller, "Transmission of IPv6 Packets over Frame Relay Networks Specification", RFC 2590, May 1999.

[RFC3146] Fujisawa, K. and A. Onoe, "Transmission of IPv6 Packets over IEEE 1394 Networks", RFC 3146, October 2001.

[RFC3572] Ogura, T., Maruyama, M., and T. Yoshida, "Internet Protocol Version 6 over MAPOS (Multiple Access Protocol Over SONET/SDH)", RFC 3572, July 2003.

[RFC4338] DeSanti, C., Carlson, C., and R. Nixon, "Transmission of IPv6, IPv4, and Address Resolution Protocol (ARP) Packets over Fibre Channel", RFC 4338, January 2006.

[RFC4391] Chu, J. and V. Kashyap, "Transmission of IP over InfiniBand (IPoIB)", RFC 4391, April 2006.

[RFC5121] Patil, B., Xia, F., Sarikaya, B., Choi, JH., and S. Madanapalli, "Transmission of IPv6 via the IPv6 Convergence Sublayer over IEEE 802.16 Networks", RFC 5121, February 2008.

[RFC5072] Varada, S., Haskins, D., and E. Allen, "IP Version 6 over PPP", RFC 5072, September 2007.

[RFC6282] Hui, J. and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks", RFC 6282, September 2011.

[RFC4944] Montenegro, G., Kushalnagar, N., Hui, J., and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks", RFC 4944, September 2007.

[RFC6775] Shelby, Z., Chakrabarti, S., Nordmark, E., and C. Bormann, "Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", RFC 6775, November 2012.

8.2. Informative References

[I-D.ietf-6man-ipv6-address-generation-privacy]
Cooper, A., Gont, F., and D. Thaler, "Privacy Considerations for IPv6 Address Generation Mechanisms", draft-ietf-6man-ipv6-address-generation-privacy-01 (work in progress), February 2014.

[Microsoft]
Davies, J., "Understanding IPv6, 3rd. ed", page 83, Microsoft Press, 2012, <http://it-ebooks.info/book/1022/>.

Authors' Addresses

Fernando Gont
SI6 Networks / UTN-FRH
Evaristo Carriego 2644
Haedo, Provincia de Buenos Aires 1706
Argentina

Phone: +54 11 4650 8472
Email: fgont@si6networks.com
URI: http://www.si6networks.com

Alissa Cooper
Cisco
707 Tasman Drive
Milpitas, CA 95035
US

Phone: +1-408-902-3950
Email: alcoop@cisco.com
URI: https://www.cisco.com/

Dave Thaler
Microsoft
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

Phone: +1 425 703 8835
Email: dthaler@microsoft.com

Will Liu
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
P.R. China

Email: liushucheng@huawei.com

6lo
Internet-Draft
Intended status: Standards Track
Expires: March 29, 2015

G. Rizzo, Ed.
AJ. Jara, Ed.
A. Olivieri
Y. Bocchi
HES-SO
MR. Palattella
SnT/Univ. of Luxembourg
L. Ladid
SnT/Univ. of Luxembourg/IPv6 Forum
S. Ziegler
C. Crettaz
Mandat International
September 25, 2014

IPv6 mapping to non-IP protocols
draft-rizzo-6lo-6legacy-02

Abstract

IPv6 is an important enabler of the Internet of Things, since it provides an addressing space large enough to encompass a vast and ubiquitous set of sensors and devices, allowing them to interconnect and interact seamlessly. To date, an important fraction of those devices is based on networking technologies other than IP. An important problem to solve in order to include them into an IPv6-based Internet of Things, is to define a mechanism for assigning an IPv6 address to each of them, in a way which avoids conflicts and protocol aliasing.

The only existing proposal for such a mapping leaves many problems unsolved and it is nowadays inadequate to cope with the new scenarios which the Internet of Things presents. This document defines a mechanism, 6TONon-IP, for assigning automatically an IPv6 address to devices which do not support IPv6 or IPv4, in a way which minimizes the chances of address conflicts, and of frequent configuration changes due to instability of connection among devices. Such a mapping mechanism enables stateless autoconfiguration for legacy technology devices, allowing them to interconnect through the Internet and to fully integrate into a world wide scale, IPv6-based IoT.

Status of This Memo

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

This Internet-Draft will expire on March 29, 2015.

Table of Contents

1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

2. Introduction

The Future Internet and the IPv6 protocol enable a new generation of techniques for accessing the network, which extend the Internet seamlessly to personal devices, sensors, home appliances, enabling the so called ''Internet of Things'' (IoT). One of the key issues which presently hampers the development of IoT and limits its potential is the lack of an efficient common framework for the integration among the vast and diverse set of protocols and technologies which compose it. Current sensors and their application environments employ a large set of technologies which lack efficient interoperability. Some associations of manufacturers have been formed to build a common technological framework in specific application domains, e.g. KNX for building automation (http://www.knx.org/), ZigBee (ZigBee Alliance) (http://www.zigbee.org/), and protocols such as X10 and CAN. Such frameworks are based on very different architectures, and the protocols which compose them are generally not interoperable. Finally, most of these technologies were designed in a context of small and local networks, with limited capabilities, and they were not conceived for integration within the Internet. One of the ideas at the basis of the IoT is the constitution of a common set of protocols which enables the interaction between devices through the Internet. By enabling interaction through the Internet, new services could be conceived and implemented, increasing the value produced by the IoT infrastructure. The adoption of a common framework may make more economically convenient its deployment, and foster the development of new smart environments (buildings, cities, etc), ultimately making possible the full realization of the potential of the IoT. As deployment of new sensors is typically expensive, it is unthinkable of putting to disuse an installed set of sensors, once a new set of devices (typically, IPv6 enabled) is deployed. This is not an uncommon case, as the set of deployed legacy devices (sensors, actuators) is to date very large. Rather, mechanisms are needed to integrate legacy devices into a common IoT platform, in order to include them in all the present and future services (e.g. devices and services directory, localization services, etc) which will be implemented on the IoT. For these reasons, many designers of the Internet of Things are focusing on building such common access and communications framework. All the proposals (e.g. CoAP, RESTful Web services) presently under discussion are based on IPv6. This has important implications on the addressing of the devices. Indeed a

common addressing at the device level is mandatory, in order to implement true Machine to Machine (M2M) communications without Portal Servers, which would make the whole system difficult to integrate and scale. The present document focuses on the network layer aspects of such IPv6 based integration. At the network layer, a mechanism which assigns an IPv6 address to each device is needed, to solve the addressing problem. In this document, we propose a new mechanism for the users and devices to map the different addressing spaces to a common IPv6 one. Our proposed mechanism solves several issues posed by some of the mappings adopted so far. Such mapping makes it possible for every device from each technology to operate through a common framework based on IPv6 and protocols over IPv6 such as RESTFul WebServices and Constrained Application Protocol (CoAP). For each technology, the proposed mechanism maps technology-specific features to a set of fields defined within the IPv6 address. This allows the location and identification of the devices in a multi-protocol card, or in any gateway or Portal Server.

2.1. Examples

In this subsection, we present two examples which help understanding the importance of adopting a common IPv6 based framework for interaction between things, and the need for legacy devices to be individually addressable through IPv6.

2.1.1. Example 1 - Building automation systems and IoT

The IoT is composed by a very large set of devices, which is poised to grow exponentially in the near future. For this reason, a directory service is needed, which offers the possibility to individuate a specific device or set of devices, with given capabilities or within a given geographical region. Let us assume such directory lists devices with their IPv6 addresses, and their function (say a temperature sensor, or a mobile phone, etc). For instance, let us consider the case of someone willing to build a map of temperatures in a given geographical region. Such directory service would allow retrieving the list of available devices within that region, each with its own IPv6 address. Assume some of those devices are legacy, non IP based temperature sensors and part of a given building automation system. Assume also that such system manages several of those temperature sensors. Even if such system would be reachable via IP, without having those sensors individually listed in the directory and appearing as ''autonomous'' things, which can be polled directly, one should resort to techniques for retrieving the temperature reading of those sensors which are specific of that building automation technology. This would make more complex the implementation of such a temperature map.

Instead, by having the building automation system expose each sensor as an IPv6 enabled device, the whole set of temperature sensors would be accessible in a homogeneous way, greatly simplifying the task.

2.1.2. Example 2 - KNX and demand-side management

KNX is a standardized (EN 50090, ISO/IEC 14543), OSI-based network communications protocol for intelligent buildings. Among the devices typically managed through KNX, we find:

- Lighting control systems;
- Heating/ventilation and air conditioning devices;
- Shutter/blind and shading control systems; and
- Energy management and electricity/gas/water metering devices.

KNX devices do not support IP. Therefore, in order to connect a KNX home network to the Internet, a gateway (KNXnet/IP router) is necessary. Other technologies for home automation are available nowadays, in which each smart device (air conditioners, washing machines, etc) supports IPv6. Let us consider a scenario in which an utility company offers an agreement to a fraction of its clients. In exchange for a cut on the energy bill, the utility company gains direct control over some appliances at the premises of the client. In this way, by powering off some of those devices in periods when the production cost of power are very high, the utility company realizes potentially high savings.

In order to implement this, the utility company sends commands to a set of devices under its direct control. For recently installed devices, the utility can assume that they support IPv6, and some application layer protocols such as CoAP. Therefore a command to switch off a device would use the IPv6 address to identify the device, and the application layer protocol to send the actual command. But for KNX devices, the command should have another format: the IPv6 address should be the one of the router bridging the IPv6 and the KNX networks, and upper layers protocols should take care of identifying the specific device inside the KNX home network to whom the command should be sent. Having to format a specific query for each specific home automation protocol adds a level of complexity which translates into higher costs of implementation and maintenance of such a service.

3. Reference System

In this section we describe a reference system where the IPv6 mapping is used. Such a system includes:

1. A set of networks running non-IPv6-compatible technologies, each with one or more hosts connected. Such networks generally use different OSI layer 3 protocols, or they may adopt a technology which does not have any layer 3 protocol.

2. A proxy, which hosts the IPv6 mapping functionality. Such device is typically connected to each of the legacy protocols networks, and it accesses the Internet via the IPv6 protocol. Such IPv6 addressing proxy performs all the necessary conversions and adaptations between IPv6 and the (local) networking protocol of the legacy technologies, in a way which depends on the specific legacy technology considered. This proxy makes use of the IPv6 mapping mechanism in order to transforms the native addressing to IPv6 Host ID and vice versa in a way that depends on the legacy technology.

Though in what follows we will describe the proposed mapping with reference to such a system, the main ideas behind it are more general, and they apply to settings others than the one of reference presented here.

4. Issues addressed through the 6TONon-IP mapping mechanism

In this section we highlight the main open issues regarding assignment of IPv6 addresses to devices which do not support IPv6 or IPv4, and we describe a set of desirable properties for a mechanism for automatic assignment of IPv6 addresses to such devices, which we name henceforth 6TONon-IP. In Appendix A of RFC 4291, a method is described for creating modified EUI-64 format Interface Identifiers out of links or nodes with IEEE EUI-64 Identifiers, or with IEEE 802 48-bit MACs. Moreover, for technologies having other link layer interface identifier, some possible mapping methods are sketched, leaving for each legacy protocol the possibility to define its own mapping method.

In the present document, we propose a mapping mechanism which enables stateless address autoconfiguration for legacy technologies, and which exploits some protocol specific identifier such as link layer interface identifiers, and the like. The proposed mapping mechanism addresses the following issues:

1. Protocol identification: For the legacy protocols to which the mapping described in RFC 4291 does not apply, a mechanism is

needed to map an IPv6 address to the right legacy protocol. This feature is necessary in case of devices which operate as proxy for more than one legacy technology at the same time.

2. Inter protocol aliasing: Without a mechanism for identifying the legacy protocol from the host part of the IPv6 address, address conflicts are possible among devices belonging to different legacy protocols. For instance, this may happen when the link layer interface identifier is the same for two devices belonging to different technologies. As several legacy technologies are characterized by a small addressing space, address conflicts are not so unlikely.

3. Conflicts between IPv6 mapped legacy technology addresses and addresses derived from (modified or not) EUI-64 format interface identifiers.

4. Intra-protocol aliasing: As several legacy technologies are characterized by a small addressing space, it is not unlikely to have two legacy devices, mapped to IPv6 addresses with the same network ID (for instance, in the case in which they belong to two separate networks of the same technology, both connected to a same proxy), and with a same interface identifier, and mapping therefore to a same IPv6 address.

Moreover, the following is a list of desirable properties for a 6TONon-IP mapping:

1. Consistency: A host should get the same IPv6 address every time it connects to a same legacy network, assuming that the configuration of all the other devices in that network remains unchanged. This allows avoiding to advertise a new address every time the host reconnects. This feature might be particularly important for devices which are not always "on", or which are not permanently connected.

2. Local Uniqueness: For devices which have an IPv6 address with a same network part, the host part should be unique for each host. This property allows avoiding address conflicts.

3. Uniqueness within the whole Internet: Coherently with the IoT vision, the host part of an IPv6 address associated to a host should be unique within the whole Internet.

Depending on the specific legacy protocol, there might be protocol specific limitations to the satisfaction of these properties. In particular, for those protocols which do not have an interface identifier which is unique, properties 1) and 2) cannot be fully

satisfied. Indeed, no mapping can solve address conflicts which take place inside a legacy protocol network. When legacy protocols have a interface identifier which is unique, this can be used to produce a unique host part of an IPv6 address, and its uniqueness would guarantee the satisfaction of properties 1), 2) and 3).

5. 6TONon-IP Mapping Method

In this section we describe the proposed strategy for forming IPv6 addresses from legacy protocol information, and the address format that derives from it. We assume that (one or more) 64 bits Network ID prefixes are given to the mapping function, which therefore computes the 64 bits of the Host ID part of the address (IPv6 interface identifier), in order to form a full IPv6 address.

The input of the proposed mapping function consists in the interface identifier of the legacy protocol.

In the proposed mapping method, the resulting Host ID part (IPv6 interface identifier) is composed by six fields, as shown in Figure 1:

- A Technology ID field (11 bits), containing a code which identifies the specific legacy protocol. This field is split into two parts, one of 6 bits, and another of 5 bits.

- U/L bit (1 bit), in order to keep compatibility with the mapping EUI-64 [RFC4291]. The U/L bit is the seventh bit of the first byte and is used to determine whether the address is universally or locally administered. This bit is set to "0", in order to indicate local scope, analogously to what proposed in [RFC4291]. This choice prevents address conflicts with IPv6 interface identifier generated from IEEE EUI-64 identifiers or IEEE 48-bit MAC identifiers.

- A Reserved field (4 bits). This field can be used in the future for the identification of different interfaces for a same technology (in the same subnetwork).

- Technology Mapping field (32 bits), which maps the interface identifier of the legacy protocol. For those protocols for which the IID is not larger than 32 bits, this field contains the 32 bits of the IID. For IID which are larger than 32 bits, a hashing function is used instead of direct mapping. In particular, some hashing algorithms such as CRC-32 are suggested. Hashing satisfies the requirements of consistency and uniqueness within a subnet with a very high probability, which depends on the hashing

algorithm used. This field is split into two parts, one of 8 bits, and another of 24 bits.

- The fourth and fifth bytes are both set to to "0x00", in order not to conflict with EUI-64 interface identifiers.

The resulting format of the Host ID part of the IPv6 address obtained from the mapping is indicated in Figure 1.

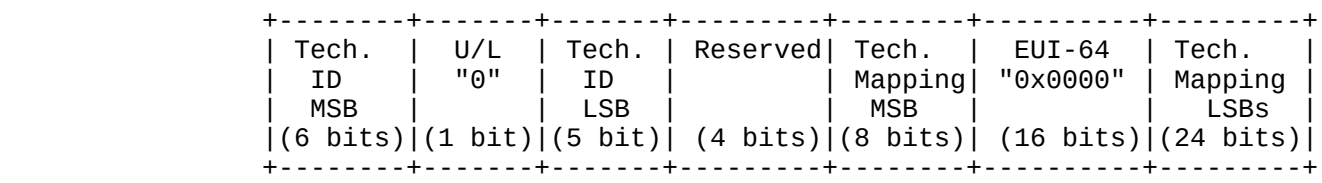


Figure 1: general format of the host ID part for legacy protocols

6. Examples

In this section we illustrate the proposed mapping method by applying it on some examples.

6.1. Example 1 - EIB/KNX

We assume the legacy protocol is EIB/KNX. This device has two kind of addresses: On the one hand, a logical address for management of group operations, and on the other hand, an individual address for identification of the device in the topology.

The mapping will be focused for the individual address. This includes an Area ID (4 bits), Line ID (8 bits), and Device ID (8 Bits). An example, is the value 0x1/0x01/0x01 for a sensor connected in the Area ID 0x1, Line ID 0x01, and Device ID 0x01.

We apply a hash (CRC-32) to the sequence 0x10101. The result is 0xDEA258A5.

Let us assume that EIB/Konnex Technology ID is "0". Thereby, the IPv6 interface identifier is "0000:DE00:00A2:58A5", considering the documentation network 2001:db8::/32. The final IPv6 address for the legacy device is "2001:db8::DE00:A2:58A5".

The address is presented in the Figure 2.

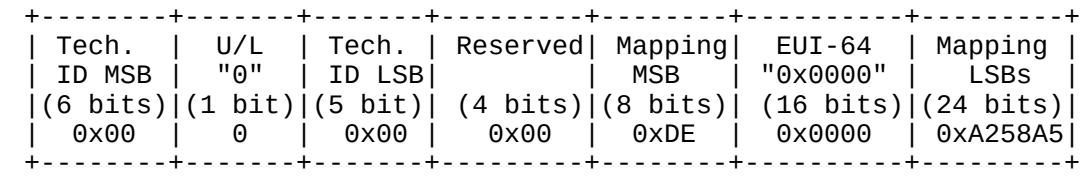


Figure 2: EIB/KNX example: the IPv6 interface identifier.

6.2. Example 2 - RFID

We assume the legacy protocol is RFID. Each RFID device is identified by its Electronic Product Code (EPC), whose length may vary from 96 to 256 bits. Let us assume to have an RFID device whose EPC is given by 01.23F3D00.8666A3.000000A05 (12 bytes). Let us assume that the RFID technology ID is "1".

We apply a hash (CRC-32) to the sequence 0x0123F3D008666A3000000A05. The result is 0xA93AFFA0.

Thereby, the IPv6 interface identifier is "0004:A900:003A:FFA0", considering the documentation network 2001:db8::/32. The final IPv6 address for the RFID tag is "2001:db8::400:A900:3A:FFA0".

The address is presented in the Figure 2.

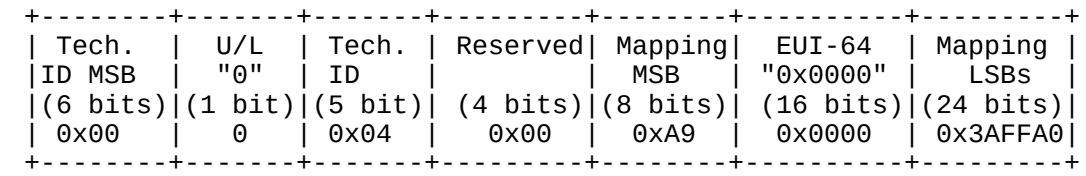


Figure 3: RFID example: the IPv6 interface identifier.

7. IANA Considerations

Not yet defined.

8. Security considerations

The proposed mapping mechanism, being based on mapping proprietary protocol ID, results in such ID being incorporated in the final IPv6 address, exposing this piece of information to the Internet. The concern has been that a user might not want to expose the details of the system to outsiders. For such concern, which holds also for MAC address mapping into EUI64 addresses, please refer to appendix B in [RFC4942].

9. Acknowledgements

The authors wish to acknowledge the following for their review and constructive criticism of this proposal: Robert Cragie. Thanks to the IoT6 European Project (STREP) of the 7th Framework Program (Grant 288445), and the colleagues who have collaborated in this work. In particular, Antonio Skarmeta from the University of Murcia, Peter Kirstein and Socrates Varakliotis from the University Colleague London, and Sebastien Ziegler from Mandat International.

10. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

[SENSORS] Jara, A., Moreno-Sanchez, P., Skarmeta, A., Varakliotis, S., and P. Kirstein,, "IPv6 Addressing Proxy: Mapping Native Addressing from Legacy Technologies and Devices to the Internet of Things (IPv6)", Sensors 13, no. 5, 6687-6712, 2013, 2013.

Authors' Addresses

Gianluca Rizzo, Ed.
HES-SO Valais
Technopole 3
Sierre, Valais 3960
Switzerland

Phone: +41-76-6151758
Email: gianluca.rizzo@hevs.ch

Antonio J. Jara, Ed.
HES-SO Valais
Technopole 3
Sierre, Valais 3960
Switzerland

Email: jara@ieee.org

Alex C. Olivieri
HES-SO Valais
Technopole 3
Sierre, Valais 3960
Switzerland

Email: Alex.Olivieri@hevs.ch

Yann Bocchi
HES-SO Valais
Technopole 3
Sierre, Valais 3960
Switzerland

Email: yann.bocchi@hevs.ch

Maria Rita Palattella
University of Luxembourg
4, rue Alphonse Weicker
Interdisciplinary Centre for Security, Reliability and Trust
Luxembourg

Phone: (+352) 46 66 44 5841
Email: maria-rita.palattella@uni.lu

Latif Ladid
University of Luxembourg / IPv6 Forum
4, rue Alphonse Weicker
Interdisciplinary Centre for Security, Reliability and Trust
Luxembourg

Phone: (+352) 46 66 44 5720
Email: latif@ladid.lu

Sebastien Ziegler
Mandat International
3 rue Champ Baron
1209 Geneva
Switzerland

Email: sziegler@mandint.org

Cedric Crettaz
Mandat International
3 rue Champ Baron
1209 Geneva
Switzerland

Email: iot6@mandint.org

6lo
Internet-Draft
Intended status: Standards Track
Expires: April 13, 2015

B. Sarikaya, Ed.
Huawei USA
F. Xia
Huawei Technologies Co., Ltd.

October 10, 2014

Lightweight and Secure Neighbor Discovery for Low-power and Lossy Networks
draft-sarikaya-6lo-cga-nd-01

Abstract

Modifications to 6lowpan Neighbor Discovery protocol are proposed in order to secure the neighbor discovery for low-power and lossy networks. This document defines lightweight and secure version of the neighbor discovery for low-power and lossy networks. The nodes generate a Cryptographically Generated Address, register the Cryptographically Generated Address with a default router and periodically refresh the registration. Cryptographically generated address and digital signatures are calculated using elliptic curve cryptography, so that the cryptographic operations are suitable for low power devices.

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 http://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 April 13, 2015.

Table of Contents

1. Introduction

Neighbor discovery for IPv6 [RFC4861] and stateless address autoconfiguration [RFC4862], together referred to as neighbor discovery protocols (NDP), are defined for regular hosts operating with wired/wireless links. These protocols are not suitable and require optimizations for resource constrained, low power hosts operating with lossy wireless links. Neighbor discovery optimizations for 6lowpan networks include simple optimizations such as a host address registration feature using the address registration option which is sent in unicast Neighbor Solicitation (NS) and Neighbor Advertisement (NA) messages [RFC6775].

Neighbor discovery protocols (NDP) are not secure especially when physical security on the link is not assured and vulnerable to attacks defined in [RFC3756]. Secure neighbor discovery protocol (SEND) is defined to secure NDP [RFC3971]. Cryptographically generated addresses (CGA) are used in SEND [RFC3972]. SEND mandates the use of the RSA signature algorithm which is computationally heavy

and not suitable to use for low-power and resource constrained nodes. The use of an RSA public key and signature leads to long message sizes not suitable to use in low-bit rate, short range, asymmetric and non-transitive links such as IEEE 802.15.4.

In this document we extend the 6lowpan neighbor discovery protocol with cryptographically generated addresses. The nodes generate CGAs and register them with the default router. CGA generation is based on elliptic curve cryptography (ECC)and signature is calculated using elliptic curve digital signature algorithm (ECDSA) known to be lightweight, leading to much smaller packet sizes. The resulting protocol is called Lightweight Secure Neighbor Discovery Protocol (LSEND).

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

The terminology in this document is based on the definitions in [RFC3971], [RFC3972] in addition to the ones specified in [RFC6775].

3. Problem Statement

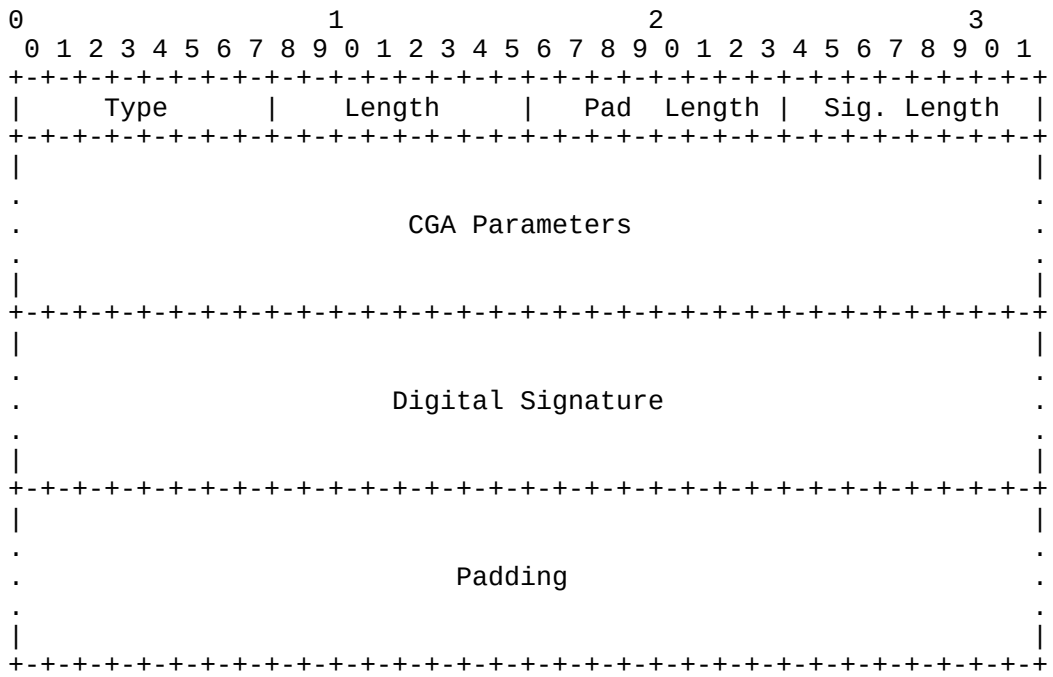
6LowPAN neighbor discovery protocol [RFC6775] needs to be extended to make it secure and also for being more efficient as well as other use cases. Requirements on such enhancements are stated in [I-D.thubert-6lo-rfc6775-update-reqs].

4. New Options

4.1. CGA Parameters and Digital Signature Option

This option contains both CGA parameters and the digital signature.

A summary of the CGA Parameters and Digital Signature Option format is shown below.



Type

TBA1 for CGA Parameters and Digital Signature

Length

The length of the option (including the Type, Length, Pad Length, Signature Length, CGA Parameters, Digital Signature and Padding fields) in units of 8 octets.

Pad Length

The length of the Padding field.

Sig Length

The length of the Digital Signature field.

CGA Parameters

The CGA Parameters field is variable-length containing the CGA Parameters data structure described in Section 4 of [RFC3972].

Digital Signature

The Digital Signature field is a variable length field containing a Elliptic Curve Digital Signature Algorithm (ECDSA) signature (with SHA-256 and P-256 curve of [FIPS-186-3]). Digital signature is constructed as explained in Section 4.3.

Padding

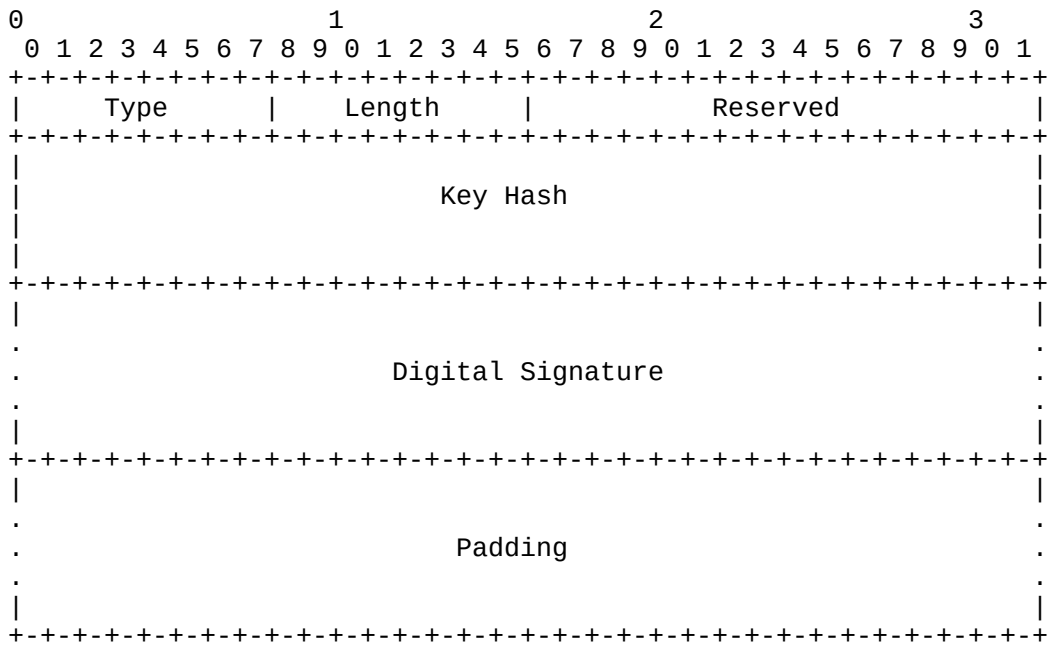
The Padding field contains a variable-length field making the CGA Parameters and Digital Signature Option length a multiple of 8.

4.2. Digital Signature Option

This option contains the digital signature.

A summary of the Digital Signature Option format is shown below. Note that this option has the same format as RSA Signature Option defined in [RFC3971]. The differences are that Digital Signature field carries an ECDSA signature not an RSA signature, and in calculating Key Hash field SHA-2 is used instead of SHA-1.

In the sequence of octets to be signed using the sender’s private key includes 128-bit CGA Message Type tag. In LSEND, CGA Message Type tag of 0xE8C47FB7FD2BB885DAB2D31A0F2808B4 MUST be used.



Type

TBA2 for Digital Signature

Length

The length of the option (including the Type, Length, Reserved, Key Hash, Digital Signature and Padding fields) in units of 8 octets.

Key Hash

The Key Hash field is a 128-bit field containing the most significant (leftmost) 128 bits of a SHA-2 hash of the public key used for constructing the signature. This is the same as in [RFC3971] except for SHA-1 which has been replaced by SHA-2.

Digital Signature

Same as in Section 4.1.

Padding

The Padding field contains a variable-length field containing as many bytes long as remain after the end of the signature.

4.3. Calculation of the Digital Signature and CGA Using ECC

Due to the use of Elliptic Curve Cryptography, the following modifications are needed to [RFC3971] and [RFC3972].

The digital signature is constructed by using the sender's private key over the same sequence of octets specified in Section 5.2 of [RFC3971] up to all neighbor discovery protocol options preceding the Digital Signature option containing the ECC-based signature. The signature value is computed using the ECDSA signature algorithm as defined in [SEC1] and hash function SHA-256.

Public Key is the most important parameter in CGA Parameters defined in Section 4.1. Public Key MUST be DER-encoded ASN.1 structure of the type SubjectPublicKeyInfo formatted as ECC Public Key. The AlgorithmIdentifier, contained in ASN.1 structure of type SubjectPublicKeyInfo, MUST be the (unrestricted) id- ecPublicKey algorithm identifier, which is OID 1.2.840.10045.2.1, and the subjectPublicKey MUST be formatted as an ECC Public Key, specified in Section 2.2 of [RFC5480].

Note that the ECC key lengths are determined by the namedCurves parameter stored in ECParameters field of the AlgorithmIdentifier. The named curve to use is secp256r1 corresponding to P-256 which is OID 1.2.840.10045.3.1.7 [SEC2].

ECC Public Key could be in uncompressed form or in compressed form where the first octet of the OCTET STRING is 0x04 and 0x02 or 0x03, respectively. Point compression using secp256r1 reduces the key size by 32 octets. In LSEND, point compression MUST be supported.

5. Protocol Interactions

Lightweight Secure Neighbor Discovery for Low-power and Lossy Networks (LSEND for LLN) modifies Neighbor Discovery Optimization for Low-power and Lossy Networks [RFC6775] as explained in this section. Protocol interactions are shown in Figure 1.

6LoWPAN Border Routers (6LBR) send router advertisements (RA). 6LoWPAN Nodes (6LN, or simply "nodes") receive these RAs and generate their own cryptographically generated addresses using elliptic curve cryptography as explained in Section 4.3. The node sends a neighbor solicitation (NS) message with the address registration option (ARO) to 6LBR. Such a NS is called an address registration NS.

An LSEND for LLN node MUST send an address registration NS message after adding CGA Parameters and Digital Signature Option defined in Section 4.1. Source address MUST be set to its crypotographically generated address. An LSEND for LLN node MUST set the Extended Unique Identifier (EUI-64) field [Guide] in ARO to the rightmost 64 bits of its crypotographically generated address. The Subnet Prefix field of CGA Parameters MUST be set to the leftmost 64 bits of its crypotographically generated address. The Public Key field of CGA Parameters MUST be set to the node's ECC Public Key.

6LBR receives the address registration NS. 6LBR then verifies the source address as described in Section 5.1.2. of [RFC3971] using the claimed source address and CGA Parameters field in the message. After successfully verifying the address 6LBR next does a cryptographic check of the signature included in the Digital Signature field in the message. If all checks succeed then 6LBR performs a duplicate address detection procedure on the address. If that also succeeds 6LBR registers the CGA in the neighbor cache. 6LBR also caches the node's public key.

6LBR sends an address registration neighbor advertisement (NA) as a reply to confirm the node's registration. Status is set to 0 to indicate success. This completes initial address registration. The address registration needs to be refreshed after the neighbor cache entry times out.

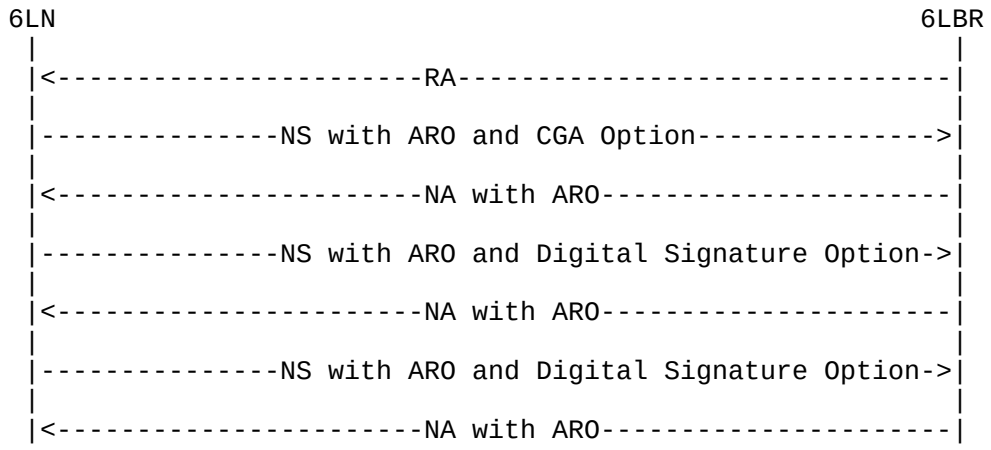


Figure 1: Lightweight SEND for LLN Protocol

In order to refresh the neighbor cache entry, an LSEND for LLN node MUST send an address registration NS message after adding the Digital

Signature Option defined in Section 4.2. The Key Hash field is a hash of the node's public key and MUST be set as described in Section 4.2. The Digital Signature field MUST be set as described in Section 4.2.

6LBR receives the address registration refresh NS. 6LBR uses the key hash field in Digital Signature Option to find the node's public key from the neighbor cache. 6LBR verifies the digital signature in the NS. In case of successful verification, 6LBR sends back an address registration neighbor advertisement (NA) to the node and sets the status to 0 indicating successful refreshment of the CGA of the node. Similar refresh NS and NA exchanges happen afterwards as shown in Figure 1.

5.1. Packet Sizes

An original address registration NS message that contains a 40 byte header and ARO is 16 octets. DER-encoded ECC Public Key for P-256 curve is 88 octets long uncompressed and 88-32=56 octets with point compression. Digital Signature field when using ECDSA for P-256 curve is 72 octets long without padding bytes for a DER encoding of the ASN.1 type "ECDSA-sig-value" [ANSIX9.62].

CGA Parameters and Digital Signature Option's CGA Parameters include 16 octet modifier, 8 octet prefix obtained from the router advertisement message sent from 6LBR, 1 octet collision count and 56 octet Public Key. Digital Signature is 72 octets. The option is 160 octets with Padding of 7 octets. The total message size of an original LSEND address registration NS message is 216 octets and such a message can be encapsulated into three 802.15.4 frames.

An address registration refresh NS message contains an ARO which is 16 octets and the digital signature option containing 16 octet key hash and 71 octet signature and 5 octet Padding. The message is 152 octets long with the header. Such a message could be encapsulated in two 802.15.4 frames.

The overhead of LSEND is valid initially and in base LSEND, possibly after bootstrapping at the address registration neighbor solicitation message. It disappears after that as we explain below in Section 6 in case optimal LSEND is used.

6. Optimizations

In this section we present optimizations to the base LSEND defined above. We use EUI-64 identifier instead of source address in CGA calculations. We also extend LSEND operation to 6LoWPAN multihop network.

Digital signature and CGA are calculated over EUI-64 or interface id of the node. It is only done initially at once not repeated with every message the node sends. The calculation does not change even if the node has a new address since EUI-64 does not change. This means that this CGA can be used to claim multiple targets. The calculation is ECC based as described in Section 4.3.

Protocol interactions are as defined in Section 5. The address registration NS message contains CGA Parameters and Digital Signature Option defined in Section 4.1. The node MUST set the Extended Unique Identifier (EUI-64) field [Guide] in ARO to the crypotographically generated address. The Subnet Prefix field of CGA Parameters MUST be set to the 64-bit prefix in the RA message received from 6LBR. Source address MUST be set to the prefix concatenated with the node's crypotographically generated address. The Public Key field of CGA Parameters MUST be set to the node's ECC Public Key.

CGA calculated may need to be modified before it is used as EUI-64. The b2 bit or U/L or "u" bit MUST be set to zero for globally unique and b1 bit or I/G or "g" bit MUST be set to zero for unicast before using it in IPv6 address as the interface identifier. In LSEND, senders and receivers ignore any differences in the three leftmost bits and in bits 6 and 7 (i.e., the "u" and "g" bits) in the interface identifiers [RFC3972].

The Target Address field in NS message is set to the prefix concatenated with the node's crypotographically generated address. This address does not need duplicate address detection as EUI-64 is globally unique. So a host cannot steal an address that is already registered unless it has the key for the EUI-64. The same EUI-64 can thus be used to protect multiple addresses e.g. when the node receives a different prefix. The node adds CGA Parameters (including Public Key) and Digital Signature Option defined in Section 4.1 into NS message. The node sends the address registration option (ARO) which is set to the CGA calculated.

Protocol interactions given in xref target="Dynamic-fig"/> are modified a bit in that Digital Signature option with the public key and ARO are passed to and stored by the 6LR/6LBR on the first NS and not sent again the in the next NS.

The 6LR/6LBR ensures first-come/first-serve by storing the ARO and the cryptographical material correlated to the target being registered. Then, if the node is the first to claim any address it likes, then it becomes owner of that address and the address is bound to the CGA in the 6LR/6LBR registry. This procedure avoids the constrained device to compute multiple keys for multiple addresses. The registration process allows the node to tie all the addresses to

the same EUI-64 and have the 6LR/6LBR enforce first come first serve after that.

6.1. Multihop Operation

In multihop 6LoWPAN, 6LBR sends RAs with prefixes downstream and it is the 6LR that receives and relays them to the nodes. 6LR and 6LBR communicate with the ICMPv6 Duplicate Address Request (DAR) and the Duplicate Address Confirmation (DAC) messages. The DAR and DAC use the same message format as NS and NA with different ICMPv6 type values.

In LSEND we extend DAR/DAC messages to carry CGA Parameters and Digital Signature Option defined in Section 4.1.

In a multihop 6LoWPAN, the node exchanges the messages shown in Figure 1 with 6LR not with 6LBR. 6LBR must be aware of who owns an address (EUI-64) to defend the first user if there is an attacker on another 6LR. Because of this the content that the source signs and the signature needs to be propagated to the 6LBR in DAR message. For this purpose we need the DAR message sent by 6LR to 6LBR MUST contain CGA Parameters and Digital Signature Option carrying the CGA that the node calculates and its public key. DAR message also contains ARO.

It is possible that occasionally, 6LR may miss the node’s CGA (that it received in ARO) or the crypto information (that it received in CGA Parameters and Digital Signature Option). 6LR should be able to ask for it again. This is done by restarting the exchanges shown in Figure 1. The result enables 6LR to refresh CGA and public key information that was lost. 6LR MUST send DAR message with CGA Parameters and Digital Signature Option and ARO to 6LBR. 6LBR as a reply forms a DAC message with the information copied from the DAR and the Status field is set to zero. With this exchange, the 6LBR can (re)validate and store the CGA and crypto information to make sure that the 6LR is not a fake.

7. Security Considerations

The same considerations regarding the threats to the Local Link Not Covered (as in [RFC3971]) apply.

The threats discussed in Section 9.2 of [RFC3971] are countered by the protocol described in this document as well.

As to the attacks to the protocol itself, denial of service attacks that involve producing a very high number of packets are deemed unlikely because of the assumptions on the node capabilities in low-power and lossy networks.

8. IANA considerations

This document defines two new options to be used in neighbor discovery protocol messages and new type values for CGA Parameters and Digital Signature Option (TBA1) and Digital Signature Option (TBA2) need to be assigned by IANA.

This document defines 0xE8C47FB7FD2BB885DAB2D31A0F2808B4 for LSEND CGA Message Type Tag.

9. Acknowledgements

Greg Zaverucha from RIM made contributions to this document. Comments from Pascal Thubert are appreciated.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3756] Nikander, P., Kempf, J., and E. Nordmark, "IPv6 Neighbor Discovery (ND) Trust Models and Threats", RFC 3756, May 2004.

[RFC3971] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, March 2005.

[RFC3972] Aura, T., "Cryptographically Generated Addresses (CGA)", RFC 3972, March 2005.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.

[RFC5480] Turner, S., Brown, D., Yiu, K., Housley, R., and T. Polk, "Elliptic Curve Cryptography Subject Public Key Information", RFC 5480, March 2009.

[RFC6775] Shelby, Z., Chakrabarti, S., Nordmark, E., and C. Bormann, "Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", RFC 6775, November 2012.

[SEC1] "Standards for Efficient Crtptography Group. SEC 1: Elliptic Curve Cryptography Version 2.0", May 2009.

[Guide] "Guidelines for 64-bit global Identifier (EUI-64TM)", November 2012, <http://standards.ieee.org/develop/regauth/tut/eui64.pdf>.

[ANSIX9.62] "American National Standards Institute (ANSI), ANS X9.62-2005: The Elliptic Curve Digital Signature Algorithm (ECDSA)", November 2005.

10.2. Informative references

[SEC2] "Standards for Efficient Crtptography Group. SEC 2: Recommended Elliptic Curve Domain Parameters Version 2.0", January 2010.

[FIPS-186-3] "National Institute of Standards and Technology, "Digital Signature Standard"", June 2009.

[NIST-ST] "National Institute of Standards and Technology, "NIST Comments on Cryptanalytic Attackts on SHA-1"", January 2009, <http://csrc.nist.gov/groups/ST/hash/statement.html>.

[I-D.rafiee-6man-ssas] Rafiee, H. and C. Meinel, "A Simple Secure Addressing Scheme for IPv6 AutoConfiguration (SSAS)", draft-rafiee-6man-ssas-11 (work in progress), September 2014.

[I-D.thubert-6lo-rfc6775-update-reqs] Thubert, P., "Requirements for an update to 6LoWPAN ND", draft-thubert-6lo-rfc6775-update-reqs-04 (work in progress), August 2014.

Authors' Addresses

Behcet Sarikaya (editor)
Huawei USA
5340 Legacy Dr. Building 3
Plano, TX 75024

Email: sarikaya@ieee.org

Frank Xia
Huawei Technologies Co., Ltd.
101 Software Avenue, Yuhua District
Nanjing, Jiangsu 210012, China

Phone: ++86-25-56625443
Email: xiayangsong@huawei.com

6tisch
Internet-Draft
Intended status: Informational
Expires: April 30, 2015

R. Struik
Struik Security Consultancy

October 27, 2014

Observations about IPv6 Addressing
draft-struik-6lo-on-ipv6-addressing-00

Abstract

This document contains some observations on IPv6 addressing.

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 RFC 2119 [RFC2119].

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 http://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 April 30, 2015.

Copyright Notice

Table of Contents

1. Opaque identifiers

RFC7217 [RFC7217] describes a mechanism for generating opaque interface identifiers and argues that these identifiers improve security and privacy of IPv6 addresses, when compared to using modified EUI-64 address formats. The main case presented in that document is that using opaque interface identifiers, rather than fixed hardware device identifiers, thwarts attempts at correlating of host activities over time, tracking across multiple networks, and pinpointing devices that may exhibit known vulnerabilities.

There are also some down sides to adopting this opaque identifier format:

1. Use of opaque identifiers does not preclude traceability on layer 2. While this is an obvious remark, the reverse also seems to hold: if Layer 2 MAC addresses would be randomized (see, e.g., discussion on MAC address randomization at IETF-90), then derivation of IPv6 addresses using those randomized MAC adddreses (rather than the EUI-64 hardware address) would certainly serve the same purpose as the technique in RFC 7217. Moreover, IPv6 opaque addresses may trickle down to Layer 2, by deriving the randomized MAC address from the interface identifier (assumed to be at least 64-bit long). This would allow constrained nodes to derive compression benefits that would not be available if one would cut the ties between Layer 2 and Layer 3 address formats. As such, this would benefit "constrained cluster" specifications, such as RFC6282, RFC4944, and RFC 6755.

2. The algorithm in RFC 7217 for generating opaque interface identifiers RID depends on an intra-device secret key (secret_key), and some public parameters (Prefix, Net_Iface, Network_ID) and takes the form RID:=F(key, public parameters). It is noted that F() MUST be difficult to reverse, MUST not be computable without knowledge of the secret key, and should not

leak the secret key given a number of samples F(key,public parms), where parms are under the control of an adversary. The output should be at least 64 bits (and, in practice, mostly is). While the specification suggests that the secret key should, indeed, be kept secret, the specification seems to allow administrator access and depends on trustworthy bootstrapping. Since it cannot be verified outside the device whether the quantity RID and the opaque interface identifier were indeed generated as specified with a secret key unknown to any outside device, this leaves this technique open to "Big Brother"-esque manipulation. Indeed, it is not hard to see (inspired by [Surveillance]) that one could field devices, where device-internal private information could be leaked via the opaque interface identifier, no matter the good intentions: the supposedly opaque interface identifier simply serves as a so-called subliminal channel. This subliminal channel cannot be detected without close examination of the entire device implementation.

2. Security Considerations

This note illustrates that privacy is a system issue and illustrates examples where the opaque interface identifier could be turned into a subliminal channel for releasing secret information to a Big Brother agent, without means for detecting this.

3. IANA Considerations

There is no IANA action required for this document.

4. Acknowledgments

Kudos to Edward Snowden for introducing fascinating technical problems to the paranoid.

5. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC7217] Gont, F., "A Method for Generating Semantically Opaque Interface Identifiers with IPv6 Stateless Address Autoconfiguration (SLAAC)", RFC 7217, April 2014.

[RFC3972] Aura, T., "Cryptographically Generated Addresses (CGA)", RFC 3972, March 2005.

[RFC6282] Hui, J. and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks", RFC 6282, September 2011.

[RFC4944] Montenegro, G., Kushalnagar, N., Hui, J., and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks", RFC 4944, September 2007.

[RFC6755] Campbell, B. and H. Tschofenig, "An IETF URN Sub-Namespace for OAuth", RFC 6755, October 2012.

[I-D.ietf-6man-default-iids]
Gont, F., Cooper, A., Thaler, D., and W. Will, "Recommendation on Stable IPv6 Interface Identifiers", draft-ietf-6man-default-iids-01 (work in progress), October 2014.

[I-D.ietf-6man-why64]
Carpenter, B., Chown, T., Gont, F., Jiang, S., Petrescu, A., and A. Yourtchenko, "Analysis of the 64-bit Boundary in IPv6 Addressing", draft-ietf-6man-why64-07 (work in progress), October 2014.

[I-D.sarikaya-6lo-cga-nd]
Sarikaya, B. and F. Xia, "Lightweight and Secure Neighbor Discovery for Low-power and Lossy Networks", draft-sarikaya-6lo-cga-nd-01 (work in progress), October 2014.

[Surveillance]
Mihir Bellare, Kenneth G. Paterson, Phillip Rogaway, "Security of Symmetric Encryption Against Mass-Surveillance", CRYPTO 2014, IACR, August 2014.

Author’s Address

Rene Struik
Struik Security Consultancy

Email: rstruik.ext@gmail.com

6Lo
Internet-Draft
Intended status: Standards Track
Expires: April 28, 2015

P. Thubert, Ed.
cisco

October 27, 2014

Requirements for an update to 6LoWPAN ND
draft-thubert-6lo-rfc6775-update-reqs-05

Abstract

Work presented at the ROLL, 6lo, 6TiSCH and 6MAN Working Groups suggest that enhancements to the 6LoWPAN ND mechanism are now needed. This document elaborates on those requirements and suggests approaches to serve them.

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 http://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 April 28, 2015.

Table of Contents

1. Introduction

A number of use cases, including the Industrial Internet, require a large scale deployment of sensors that can not be realized with wires and is only feasible over wireless Low power and Lossy Network (LLN) technologies. When simpler hub-and-spoke topologies are not sufficient for the expected throughput and density, mesh networks must be deployed, which implies the concepts of hosts and routers, whether operated at Layer-2 or Layer-3.

The IETF has designed the LLN host-to-router and router-to-router protocol that supports address assignment and the router-to-router protocol that supports reachability across Route-Over LLNs in different Areas. It was clear for both efforts that the scalability requirements could only be met with IPv6 [RFC2460], and there is no fundamental contradiction between those protocols to that regard.

While DHCPv6 is still a viable option in LLNs, the new IETF standard that supports address assignment specifically for LLNs is 6LoWPAN ND, the Neighbor Discovery Optimization for Low-power and Lossy Networks [RFC6775]. 6LoWPAN ND was designed as a stand-alone mechanism separately from its IETF routing counterpart, the IPv6 Routing Protocol for Low power and Lossy Networks [RFC6550] (RPL), and the interaction between the 2 protocols was not defined.

The 6TiSCH WG is now considering an architecture [I-D.ietf-6tisch-architecture] whereby a 6LowPAN ND host could connect to the Internet via a RPL Network, but this requires additions to the protocol to support mobility and reachability in a secured and manageable environment.

At the same time, new work at 6MAN on Efficiency aware IPv6 Neighbor Discovery Optimizations [I-D.chakrabarti-nordmark-6man-efficient-nd] suggests that 6LoWPAN ND can be extended to other types of networks on top of the Low power and Lossy Networks (LLNs) for which it was already defined. The value of such extension is especially apparent in the case of mobile wireless devices, to reduce the multicast operations that are related to classical ND ([RFC4861], [RFC4862]) and plague the wireless medium. In this context also, there is a need for additions to the protocol.

The Optimistic Duplicate Address Detection [RFC4429] (ODAD) specification details how an address can be used before a Duplicate Address Detection (DAD) is complete, and insists that an address that is TENTATIVE should not be associated to a Source Link-Layer Address Option in a Neighbor Solicitation message. As we expect the 6LoWPAN ND protocol for a more general use, it can make sense to keep respecting that rule, which is another change to the specification.

This document suggests a limited evolution to [RFC6775] so as to allow operation of a 6LoWPAN ND node as a leaf in a RPL network. It also suggests a more generalized use of the information in the ARO option outside of the strict LLN domain, for instance over a converged backbone.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

Readers are expected to be familiar with all the terms and concepts that are discussed in "Neighbor Discovery for IP version 6" [RFC4861], "IPv6 Stateless Address Autoconfiguration" [RFC4862], "IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals" [RFC4919], Neighbor Discovery Optimization for Low-power and Lossy Networks [RFC6775] and "Transmission of IPv6 Packets over IEEE 802.15.4 Networks" [RFC4944].

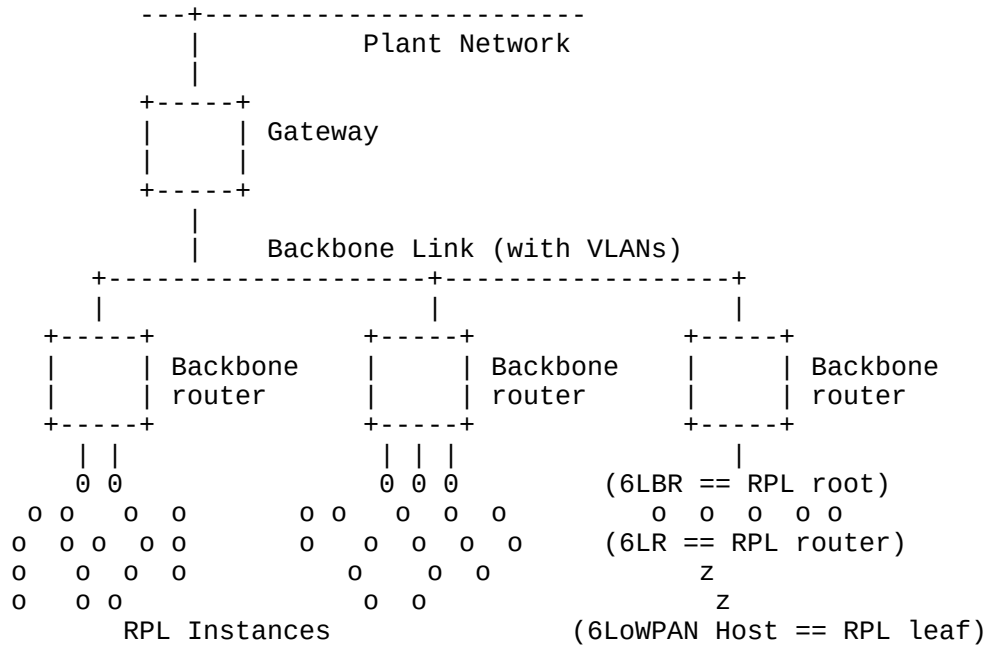
Additionally, this document uses terminology from 6TiSCH [I-D.ietf-6tisch-terminology] and ROLL [RFC7102].

3. Overview

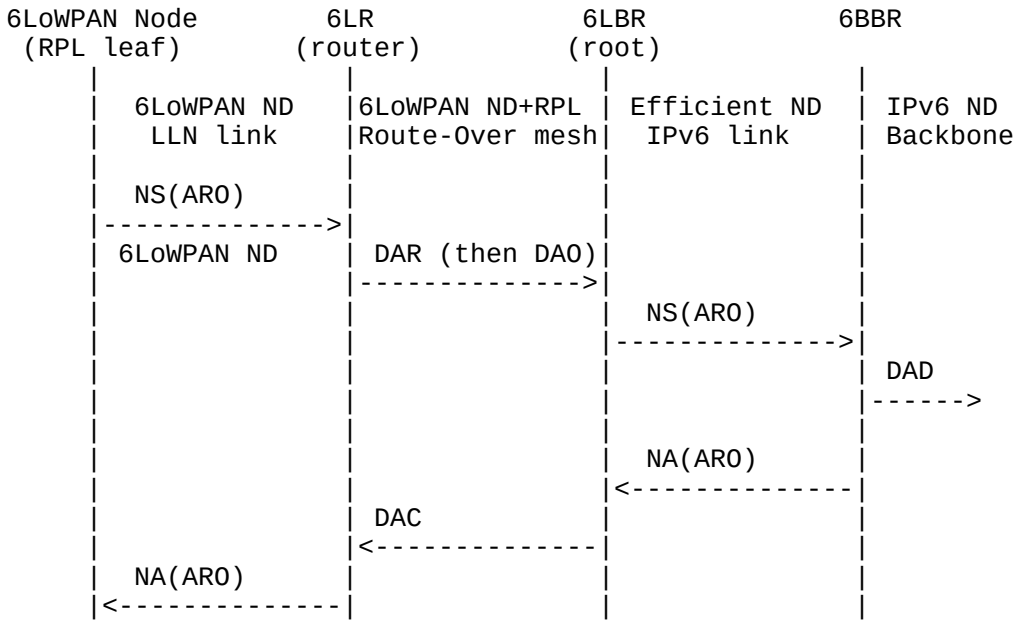
The 6TiSCH architecture [I-D.ietf-6tisch-architecture] expects that a 6LoWPAN device can connect as a leaf to a RPL network, where the leaf support is the minimal functionality to connect as a host to a RPL network without the need to participate to the full routing protocol. The support of leaf can be implemented as a minor increment to 6LoWPAN ND, with the additional capability to carry a sequence number that is used to track the movements of the device, and optionally some information about the RPL topology that this

device will join.

The scope of the 6TiSCH Architecture is a Backbone Link that federates multiple LLNs as a single IPv6 Multi-Link Subnet. Each LLN in the subnet is anchored at a Backbone Router (6BBR). The Backbone Routers interconnect the LLNs over the Backbone Link and emulate that the LLN nodes are present on the Backbone by proxy-ND operations. An LLN node can move freely from an LLN Route-Over mesh anchored at a Backbone Router to another anchored at a same or a different Backbone Router inside the Multi-Link Subnet and conserve its addresses.



The root of the RPL topology is logically separated from the 6BBR that is used to connect the RPL topology to the backbone. The RPL root can use Efficient ND as the interface to register an LLN node in its topology to the 6BBR for whatever operation the 6BBR performs, such as ND proxy operations, or injection in a routing protocol. It results that, as illustrated in Figure 2, the periodic signaling could start at the leaf node with 6LoWPAN ND, then would be carried over RPL to the RPL root, and then with Efficient-ND to the 6BBR. Efficient ND being an adaptation of 6LoWPAN ND, it makes sense to keep those two homogeneous in the way they use the source and the target addresses in the Neighbor Solicitation (NS) messages for registration, as well as in the options that they use for that process.



As the network builds up, a node should start as a leaf to join the RPL network, and may later turn into both a RPL-capable router and a 6LR, so as to accept leaf nodes to recursively join the network.

Section 5 of the 6TiSCH architecture [I-D.ietf-6tisch-architecture] provides more information on the need to update the protocols that sustain the requirements in the next section.

4. Requirements

4.1. Requirements Related to Mobility

Due to the nature of LLN networks, even a fixed 6LoWPAN Node may change its point of attachment (a 6LR) and may not be able to notify the 6LR that it has disconnected from. It results that the previous 6LR may still attract traffic that it cannot deliver any more. When the 6LR changes, there is thus a need to identify stale states and restore reachability timely.

Req1.1: Upon a change of point of attachment, connectivity via a new 6LR MUST be restored timely without the need to de-register from the previous 6LR.

Req1.2: For that purpose, the protocol MUST enable to differentiate multiple registrations from a same 6LoWPAN Node from two different 6LoWPAN Nodes claiming a same address.

Req1.3: This information MUST be passed from the 6LR to the 6LBR, and the 6LBR SHOULD be able to clean up the stale state asynchronously in the previous 6LR.

Req1.4: A 6LoWPAN Node SHOULD also be capable to register a same Address to multiple 6LRs, and this, concurrently.

4.2. Requirements Related to Routing Protocols

The point of attachment of a 6LoWPAN Node may be a 6LR in an LLN mesh. An LLN route-over mesh is typically based on RPL, which is the routing protocol that was defined at the IETF for this particular purpose. It derives that in this scenario, the 6LR would classically support RPL. One goal is that a 6LoWPAN Node attached via ND to a RPL-capable 6LR would not need to participate to the RPL protocol to obtain reachability via the 6LR. An additional goal would be to obtain reachability via other routing protocols through a same ND-based abstraction.

Related requirements are:

Req2.1: The ND registration method SHOULD be extended in such a fashion that the 6LR MAY advertise the Address of a 6LoWPAN Node over RPL and obtain reachability to that Address over the RPL domain.

Req2.2: The Address Registration Option that is used in the ND registration SHOULD be extended to carry enough information to generate a DAO message as specified in [RFC6550] section 6.4, in particular the capability to compute a DAOSequence and, as an option, a RPLInstanceID.

Req2.3: Depending on their applicability to LLNs, other standard mesh /MANET protocols MAY be considered as well.

Req2.4: Multicast operations SHOULD be supported and optimized. Groups MAY be formed by device type (e.g. routers, street lamps), location (Geography, RPL sub-tree), or both. RPL already has the capability to advertise multicast groups; whether ND is appropriate for the registration to the 6BBR is to be defined, considering the additional burden of supporting the Multicast Listener Discovery Version 2 [RFC3810] (MLDv2) for IPv6.

4.3. Requirements Related to the Variety of Low-Power Link types

6LoWPAN ND [RFC6775] was defined with a focus on IEEE802.15.4 and in particular the capability to derive a unique Identifier from a globally unique MAC-64 address. At this point, the 6lo Working Group is extending the 6LoWPAN Header Compression (HC) [RFC6282] technique to other link types ITU-T G.9959 [I-D.brandt-6man-lowpanz], Master-Slave/Token-Passing [I-D.ietf-6lo-6lobac], DECT Ultra Low Energy [I-D.ietf-6lo-dect-ule], Near Field Communication [I-D.hong-6lo-ipv6-over-nfc], as well as IEEE1901.2 Narrowband Powerline Communication Networks [I-D.popa-6lo-6loplc-ipv6-over-ieee19012-networks] and BLUETOOTH(R) Low Energy [I-D.ietf-6lo-btle].

Related requirements are:

Req3.1: The support of the registration mechanism SHOULD be extended to more LLN links, matching at least the links that are considered by 6lo as well as other popular Low-Power links such as Low-Power Wi-Fi.

Req3.2: As part of this extension, a mechanism to compute a unique Identifier should be provided, with the capability to form a Link-Local Address that can not be a duplicate. The Identifier SHOULD be unique at least to the domain where an Address formed by this device may be advertised through ND mechanisms.

Req3.3: The Address Registration Option used in the ND registration SHOULD be extended to carry the relevant forms of unique Identifier.

4.4. Requirements Related to Proxy Operations

Sleeping devices may not be able to answer themselves to a lookup from a node that uses classical ND on a backbone and may need a proxy operation by a 6BBR. Additionally, the device may need to rely on the 6LBR to perform that registration to the 6BBR.

Related requirements are:

Req4.1: The registration mechanism SHOULD enable a third party to proxy register an Address on behalf of a 6LoWPAN node that may be sleeping or located deeper in an LLN mesh.

4.5. Requirements Related to Security

In order to guarantee the operations of the 6LoWPAN ND flows, the spoofing of the 6LR, 6LBR and 6BBRs roles should be avoided. Once a node successfully registers an address, 6LoWPAN ND should provide energy-efficient means to protect that ownership even if the node is sleeping. In particular, the 6LR and the 6LBR then should be able to verify whether a subsequent registration for a same Address comes from a same node or is a duplicate.

Related requirements are:

Req5.1: 6LoWPAN ND security mechanisms SHOULD provide a mechanism for the 6LR, 6LBR and 6BBR to authenticate and authorize one another for their respective roles, as well as with the 6LoWPAN Node for the role of 6LR.

Req5.2: 6LoWPAN ND security mechanisms SHOULD provide a mechanism for the 6LR and the 6LBR to validate whether a new registration corresponds to a same 6LoWPAN Node, and, if not, determine the rightful owner, and deny or clean-up the registration that is deemed in excess.

Req5.3: 6LoWPAN ND security mechanisms SHOULD lead to small packet sizes. In particular, the NS, NA, DAR and DAC messages for a re-registration flow SHOULD NOT exceed 80 octets so as to fit in a secured IEEE802.15.4 frame.

Req5.4: Recurrent 6LoWPAN ND security operations MUST NOT be computationally intensive on the LoWPAN Node CPU. When a Key hash calculation is employed, a mechanism lighter than SHA-1 SHOULD be preferred.

Req5.5: The number of Keys that the 6LoWPAN Node needs to manipulate SHOULD be minimized.

Req5.6: The 6LoWPAN ND security mechanisms SHOULD enable CCM* for use at both Layer 2 and Layer 3, and SHOULD enable the reuse of security code that has to be present on the device for upper layer security such as TLS.

Req5.7: Public key and signature sizes SHOULD be minimized while maintaining adequate confidentiality and data origin authentication for multiple types of applications with various degrees of criticality.

4.6. Requirements Related to Low-Power devices

The ND registration method is designed to save energy on Low-Power devices, and in particular enable duty-cycled devices that are sleeping most of the time and not capable to defend their own Addresses against always-on devices.

Related requirements are:

Req6.1: The registration mechanism SHOULD be applicable to a Low-Power device regardless of the link type, and enable a 6BBR to operate as a proxy to defend the registered Addresses on its behalf.

Req6.2: The registration mechanism SHOULD enable long sleep durations, in the order of multiple days to a month, for devices capable of operating over the course of ten or more years without the need to recharge or replace the batteries.

4.7. Requirements Related to Scalability

Use cases from Automatic Meter Reading (AMR, collection tree operations) and Advanced Metering Infrastructure (AMI, bi-directional communication to the meters) indicate the needs for a large number of LLN nodes pertaining to a single RPL DODAG (e.g. 5000) and connected to the 6LBR over a large number of LLN hops (e.g. 15).

Related requirements are:

Req7.1: The registration mechanism SHOULD enable a single 6LBR to register multiple thousands of devices.

Req7.2: The timing of the registration operation should allow for a large latency such as found in LLNs with ten and more hops.

5. Security Considerations

This specification expects that the link layer is sufficiently protected, either by means of physical or IP security for the Backbone Link or MAC sublayer cryptography. In particular, it is expected that the LLN MAC provides secure unicast to/from the Backbone Router and secure broadcast from the Backbone Router in a way that prevents tempering with or replaying the RA messages. Still, Section 4.5 has a requirement for a mutual authentication and authorization for a role for 6LRs, 6LBRs and 6BBRs.

This documents also suggests in Appendix Appendix A.4 that a 6LoWPAN Node could form a single Unique Interface ID (CUID) based on cryptographic techniques similar to CGA. The CUID would be used as Unique Interface Identifier in the ARO option and new Secure ND procedures would be proposed to use it as opposed to the source IPv6 address to secure the binding between an Address and its owning Node, and enforce First/Come-First/Serve at the 6LBR.

6. IANA Considerations

This draft does not require an IANA action.

7. Acknowledgments

The author wishes acknowledge the contributions by Samita Chakrabarti, Erik Normark, JP Vasseur, Eric Levy-Abegnoli, Patrick Wetterwald, Thomas Watteyne, and Behcet Sarikaya.

8. References

8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2460] Deering, S.E. and R.M. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC3775] Johnson, D., Perkins, C. and J. Arkko, "Mobility Support in IPv6", RFC 3775, June 2004.

[RFC3810] Vida, R. and L. Costa, "Multicast Listener Discovery Version 2 (MLDv2) for IPv6", RFC 3810, June 2004.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

[RFC4429] Moore, N., "Optimistic Duplicate Address Detection (DAD) for IPv6", RFC 4429, April 2006.

[RFC4443] Conta, A., Deering, S. and M. Gupta, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 4443, March 2006.

[RFC4861] Narten, T., Nordmark, E., Simpson, W. and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, September 2007.

[RFC4862] Thomson, S., Narten, T. and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, September 2007.

[RFC4944] Montenegro, G., Kushalnagar, N., Hui, J. and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks", RFC 4944, September 2007.

[RFC6275] Perkins, C., Johnson, D. and J. Arkko, "Mobility Support in IPv6", RFC 6275, July 2011.

[RFC6282] Hui, J. and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks", RFC 6282, September 2011.

[RFC6550] Winter, T., Thubert, P., 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, March 2012.

[RFC6655] McGrew, D. and D. Bailey, "AES-CCM Cipher Suites for Transport Layer Security (TLS)", RFC 6655, July 2012.

[RFC6775] Shelby, Z., Chakrabarti, S., Nordmark, E. and C. Bormann, "Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", RFC 6775, November 2012.

[RFC6830] Farinacci, D., Fuller, V., Meyer, D. and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, January 2013.

8.2. Informative References

[I-D.brandt-6man-lowpanz]
Brandt, A. and J. Buron, "Transmission of IPv6 packets over ITU-T G.9959 Networks", Internet-Draft draft-brandt-6man-lowpanz-02, June 2013.

[I-D.chakrabarti-nordmark-6man-efficient-nd]
Chakrabarti, S., Nordmark, E., Thubert, P. and M. Wasserman, "Wired and Wireless IPv6 Neighbor Discovery Optimizations", Internet-Draft draft-chakrabarti-nordmark-6man-efficient-nd-04, October 2013.

[I-D.hong-6lo-ipv6-over-nfc]
Hong, Y., Choi, Y., Youn, J., Kim, D. and J. Choi, "Transmission of IPv6 Packets over Near Field Communication", Internet-Draft draft-hong-6lo-ipv6-over-nfc-01, August 2014.

[I-D.ietf-6lo-6lobac]
Lynn, K., Martocci, J., Neilson, C. and S. Donaldson, "Transmission of IPv6 over MS/TP Networks", Internet-Draft draft-ietf-6lo-6lobac-00, July 2014.

[I-D.ietf-6lo-btle]
Nieminen, J., Savolainen, T., Isomaki, M., Patil, B., Shelby, Z. and C. Gomez, "Transmission of IPv6 Packets over BLUETOOTH(R) Low Energy", Internet-Draft draft-ietf-6lo-btle-02, June 2014.

[I-D.ietf-6lo-dect-ule]
Mariager, P., Petersen, J., Shelby, Z., Logt, M. and D. Barthel, "Transmission of IPv6 Packets over DECT Ultra Low Energy", Internet-Draft draft-ietf-6lo-dect-ule-00, June 2014.

[I-D.ietf-6tisch-architecture]
Thubert, P., Watteyne, T. and R. Assimiti, "An Architecture for IPv6 over the TSCH mode of IEEE 802.15.4e", Internet-Draft draft-ietf-6tisch-architecture-01, February 2014.

[I-D.ietf-6tisch-terminology]
Palattella, M., Thubert, P., Watteyne, T. and Q. Wang, "Terminology in IPv6 over the TSCH mode of IEEE 802.15.4e", Internet-Draft draft-ietf-6tisch-terminology-00, November 2013.

[I-D.popa-6lo-6loplc-ipv6-over-ieee19012-networks]
Popa, D. and J. Hui, "6LoPLC: Transmission of IPv6 Packets over IEEE 1901.2 Narrowband Powerline Communication Networks", Internet-Draft draft-popa-6lo-6loplc-ipv6-over-ieee19012-networks-00, March 2014.

[RFC3610] Whiting, D., Housley, R. and N. Ferguson, "Counter with CBC-MAC (CCM)", RFC 3610, September 2003.

[RFC3963] Devarapalli, V., Wakikawa, R., Petrescu, A. and P. Thubert, "Network Mobility (NEMO) Basic Support Protocol", RFC 3963, January 2005.

[RFC3971] Arkko, J., Kempf, J., Zill, B. and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, March 2005.

[RFC3972] Aura, T., "Cryptographically Generated Addresses (CGA)", RFC 3972, March 2005.

[RFC4389] Thaler, D., Talwar, M. and C. Patel, "Neighbor Discovery Proxies (ND Proxy)", RFC 4389, April 2006.

[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, August 2007.

[RFC7102] Vasseur, JP., "Terms Used in Routing for Low-Power and Lossy Networks", RFC 7102, January 2014.

Appendix A. Suggested Changes to Protocol Elements

Appendix A.1. ND Neighbor Solicitation (NS)

The NS message used for registration should use a source address that respects the rules in [RFC6775], [RFC4861], and [RFC4429] for DAD. The SLLA Option may be present but only if the address passed DAD, and it is used to allow the 6LR to respond as opposed to as a registration mechanism.

The address that is being registered is the target address in the NS message and the TLLA Option must be present.

Appendix A.2. ND Router Advertisement (RA)

[I-D.chakrabarti-nordmark-6man-efficient-nd] adds an ’E’ bit in the Router Advertisement flag, as well as a new Registrar Address Option (RAO). These fields are probably pertinent to LLNs inclusion into a revised 6LoWPAN ND should be studied. If the new 6LoWPAN flows require a change of behaviour (e.g. registering the Target of the NS message) then the RA must indicate that the router supports the new capability, and the NS must indicate that the Target is registered as opposed to the Source in an unequivocal fashion.

There is some amount of duplication between the options in the RPL DIO [RFC6550] and the options in the ND RA messages. At the same time, there are a number of options, including the 6LoWPAN Context Option (6CO) [RFC6775], the MTU and the SLLA Options [RFC4861] that can only be found in the RA messages. Considering that these options are useful for a joining node, the recommendation would be to associate the RA messages to the join beacon, and make them rare when the network is stable. On the other hand, the DIO message is to be used as the propagated heartbeat of the RPL network and provide the sense of time and liveliness.

RAs should also be issued and the information therein propagated when a change occurs in the information therein, such as a router or a prefix lifetime.

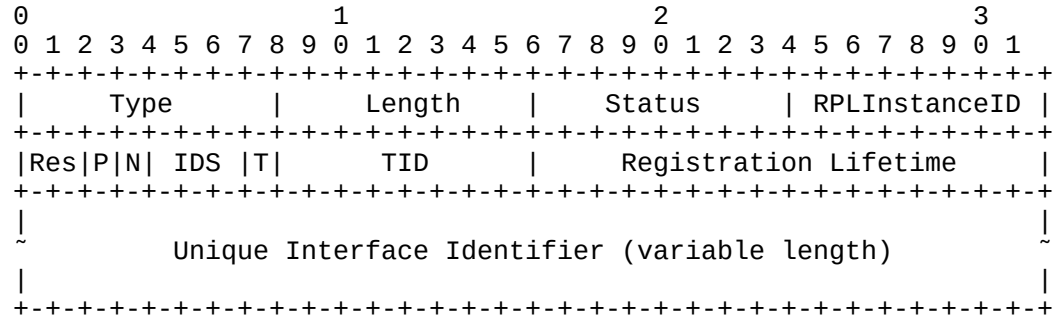
Appendix A.3. RPL DODAG Information Object (DIO)

If the RPL root serves as 6LBR, it makes sense to add at least a bit of information in the DIO to signal so. A Registrar Address Option (RAO) may also be considered for addition.

Appendix A.4. ND Enhanced Address Registration Option (EARO)

The ARO option contains a Unique ID that is supposed to identify the device across multiple registrations. It is envisioned that the device could form a single CGA-based Unique Interface ID (CUID) to securely bind all of its addresses. The CUID would be used as Unique Interface Identifier in the ARO option and to form a Link-Local address that would be deemed unique regardless of the Link type. Provided that the relevant cryptographic material is passed to the 6LBR upon the first registration or on-demand at a later time, the 6LBR can validate that a Node is effectively the owner of a CUID, and ensure that the ownership of an Address stays with the CUID that registered it first.

This option is designed to be used with standard NS and NA messages between backbone Routers as well as between nodes and 6LRs over the LLN and between the 6LBR and the 6BBR over whatever IP link they use to communicate.



The representation above is based on [I-D.chakrabarti-nordmark-6man-efficient-nd]. Only the proposed changes from that specification are discussed below but the expectation is that 6LoWPAN ND and Efficient ND converge on the ARO format.

Status: 8-bit integer. A new value of 3 is suggested to indicate a rejection due to an obsolete TID, typically an indication of a movement.

RPLInstanceID: 8-bit integer. This field is set to 0 when unused. Otherwise it contains the RPLInstanceID for which this address is registered, as specified in RPL [RFC6550], and discussed in particular in section 3.1.2.

P: One bit flag. When the bit is set, the address being registered is Target of the NS as opposed to the Source, for instance to enable ND proxy operation.

N: One bit flag. Set if the device moved. If not set, the 6BBR will refrain from sending gratuitous NA(O) or other form of distributed ND cache clean-up over the backbone. For instance, the flag should be reset after the DAD operation upon address formation.

Author’s Address

Pascal Thubert, editor
Cisco Systems, Inc
Building D
45 Allee des Ormes - BP1200
MOUGINS - Sophia Antipolis, 06254
FRANCE

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

6lo
Internet-Draft
Updates: 6282 (if approved)
Intended status: Standards Track
Expires: April 27, 2015

P. Thubert, Ed.
Cisco
C. Bormann
TZI

October 24, 2014

A compression mechanism for the RPL option draft-thubert-6lo-rpl-nhc-02

Abstract

This specification defines the RPL Packet Information (RPI) NHC compression, a method to compress RPL Option (RFC6553) information within 6LoWPAN-style ("6lo") adaptation layers. This extends 6LoWPAN Header Compression (RFC6282), saving up to 48 bits in each frame compared to the uncompressed form in RFC 6553.

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 http://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 April 27, 2015.

Copyright Notice

Table of Contents

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. The other constraints, such as the memory capacity and the duty cycling of the LLN devices, derive from that primary concern. Energy is typically available from batteries that are expected to last for years, or scavenged from the environment in very limited quantities. Any protocol that is intended for use in LLNs must be designed with the primary concern of saving energy as a strict requirement.

Controlling the amount of data transmission is one possible venue to save energy. In a number of LLN standards, the frame size is limited to much smaller values than the IPv6 maximum transmission unit (MTU) of 1280 bytes. In particular, an LLN that relies on the classical Physical Layer (PHY) of IEEE 802.14.5 [IEEE802154] is limited to 127 bytes per frame. The need to compress IPv6 packets over IEEE 802.14.5 led to the 6LoWPAN Header Compression [RFC6282] work (6LoWPAN-HC).

The Routing Protocol for Low Power and Lossy Networks [RFC6550] (RPL) is designed to optimize the routing operations in constrained LLNs.

As part of this optimization, RPL requires the addition of RPL Packet Information (RPI) in every packet, as defined in Section 11.2 of [RFC6550].

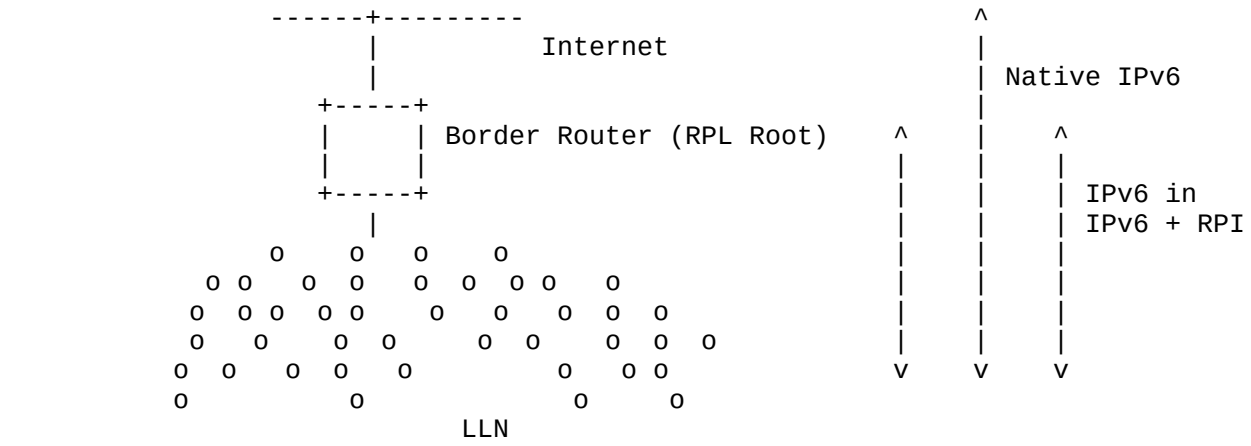


Figure 1: IP-in-IP Encapsulation within the LLN

The RPI is used to tag the packet with the RPL Instance ID and other information that RPL requires for its operation within the RPL domain. In particular, the SenderRank, which is the scalar metric computed by an specialized Objective Function such as [RFC6552], indicates the Rank of the sender and is modified at each hop. The SenderRank allows to validate that the packet progresses in the expected direction, either upwards or downwards, along the DODAG.

RPL defines the RPL Option for Carrying RPL Information in Data-Plane Datagrams [RFC6553] to transport the RPI, which is carried in an IPv6 Hop-by-Hop Options Header [RFC2460], typically consuming eight bytes per packet.

6TiSCH [I-D.ietf-6tisch-architecture] specifies the operation of IPv6 over the TimeSlotted Channel Hopping [I-D.ietf-6tisch-tsch] (TSCH) mode of operation of IEEE 802.14.5. The architecture requires the use of both 6LoWPAN HC and RPL over IEEE 802.14.5. Because it inherits the constraints on the frame size from the MAC layer, 6TiSCH cannot afford to spend 8 bytes per packet on the RPI. Hence the requirement for a 6LoWPAN header compression of the RPI.

This specification extends the 6lo adaptation layer framework ([RFC4944], [RFC6282]) to carry the same information in a 6LoWPAN RPL Packet Information (RPI) NHC Next-header compression (NHC) header, usually eliminating the Hop-by-Hop Options Header saving up to six bytes per packet.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

The Terminology used in this document is consistent with and incorporates that described in ‘Terminology in Low power And Lossy Networks’ [RFC7102] and [RFC6550].

The term "byte" is used in its now customary sense as a synonym for "octet".

3. Updating RFC 6282

This specification proposes a new 6LoWPAN Next Header Compression (NHC) for the RPL option [RFC6553], called RPI_NHC, to be placed in a packet that is compressed per [RFC6282].

It updates [RFC6282] in that the "necessary property of encoding headers using LOWPAN_NHC" becomes that "the immediately preceding header must be encoded using either LOWPAN_IPHC, RPI_NHC or LOWPAN_NHC".

Additionally, the necessary property of encoding headers using RPI_NHC is that the immediately preceding header must be encoded using either LOWPAN_IPHC or LOWPAN_NHC.

(Discuss: Is this really an update of RFC 6282 or a straightforward addition to it?)

4. The RPL Packet Information NHC

[RFC6550], Section 11.2, specifies the RPL Packet Information (RPI) as a set of fields that are to be added to the IP packets for the purpose of Instance Identification, as well as Loop Avoidance and Detection.

[RFC6553] defines an encoding for the RPI as a RPL option located in the IPv6 Hop-by-hop Option Header. The present NHC compression mechanism compresses IPv6 Hop-by-hop Headers that contain only that RPL option.

The fields in the RPI include an ’O’, an ’R’, and an ’F’ bit, an 8-bit RPLInstanceID (with some internal structure), and a 16-bit SenderRank.

This section defines the format of the RPL Packet Information NHC (RPI_NHC) that is used to compress the RPI in 6LoWPAN networks.

4.1. Compressing the RPLInstanceId

RPL Instances are discussed in [RFC6550], Section 5. A number of simple use cases will not require more than one instance, and in such a case, the instance is expected to be the global Instance 0. A global RPLInstanceID is encoded in a RPLInstanceID field as follows:

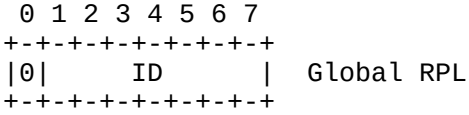


Figure 2: RPLInstanceID Field Format for Global Instances

For the particular case of the global Instance 0, the RPLInstanceID field is all zeroes. This specification allows to elide a RPLInstanceID field that is all zeroes, and defines a 'I' flag that, when set, signals that the field is elided.

4.2. Compressing the SenderRank

The SenderRank is the result of the DAGRank operation on the rank of the sender; here the DAGRank operation is defined in [RFC6550], Section 3.5.1, as:

DAGRank(rank) = floor(rank/MinHopRankIncrease)

If MinHopRankIncrease is set to a multiple of 256, the least significant 8 bits of the SenderRank will be all zeroes; by eliding those, the SenderRank can be compressed into a single byte. This idea is used in [RFC6550] by defining DEFAULT_MIN_HOP_RANK_INCREASE as 256 and in [RFC6552] that defaults MinHopRankIncrease to DEFAULT_MIN_HOP_RANK_INCREASE.

This specification allows to encode the SenderRank as either one or two bytes, and defines a 'K' flag that, when set, signals that a single byte is used.

4.3. The RPI_NHC encoding

[RFC6553] defines an encoding for the RPL information as a RPL Option located in an IPv6 Hop-by-Hop Option Header. The RPI_NHC provides a compressed form for that information and is constructed as follows:

The RPI_NHC is immediately followed by the RPLInstanceID, unless that is elided (I=1), and then the SenderRank, which is either compressed into one byte (K=1) or fully inlined as the whole 2 bytes (K=0). Bits in the RPI_NHC indicate whether the RPLInstanceID is elided and/or the SenderRank is compressed:

O, R, and F bits: The O, R, and F bits as defined in [RFC6550], Section 11.2.

NH: 1-bit flag. The Next Header (NH) bit is defined in [RFC6282], Section 4.2, and it is set to indicate that the next header is encoded using LOWPAN_NHC

I: 1-bit flag. If it is set, the Instance ID is elided and the RPLInstanceID is the Global RPLInstanceID 0. If it is not set, the byte immediately following the RPI_NHC contains the RPLInstanceID as specified in [RFC6550], Section 5.1.

K: 1-bit flag. If it is set, the SenderRank is compressed into one byte, and the least significant byte is elided. If it is not set, the SenderRank is fully inlined as 2 bytes.

In Figure 3, the RPLInstanceID is the Global RPLInstanceID 0, and the MinHopRankIncrease is a multiple of 256 so the least significant byte is all zeroes and can be elided:

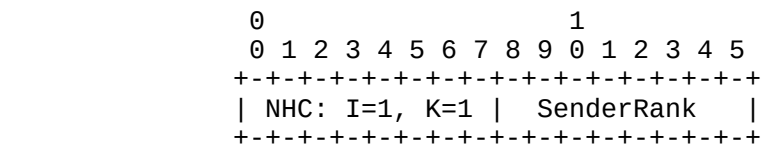


Figure 3: The most compressed RPI_NHC

In Figure 4, the RPLInstanceID is the Global RPLInstanceID 0, but both bytes of the SenderRank are significant so it can not be compressed:

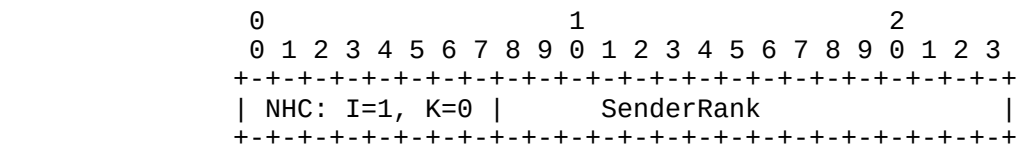


Figure 4: Eliding the RPLInstanceID

In Figure 5, the RPLInstanceID is not the Global RPLInstanceID 0, and the MinHopRankIncrease is a multiple of 256:

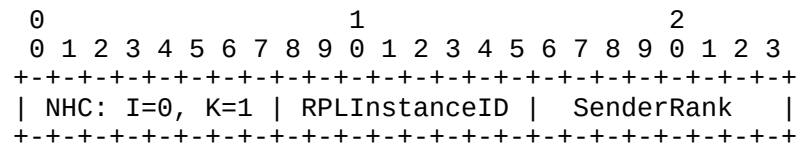


Figure 5: Compressing SenderRank

In Figure 6, the RPLInstanceID is not the Global RPLInstanceID 0, and both bytes of the SenderRank are significant:

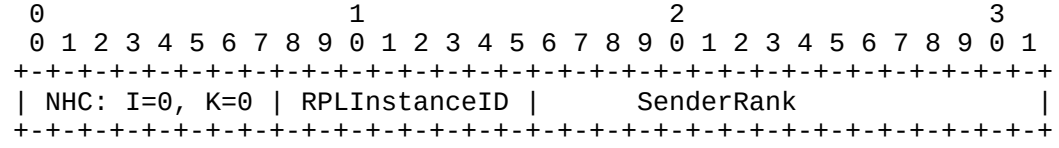


Figure 6: Least compressed form of RPI_NHC

The next sections provide alternatives for format of the RPI_NHC.

4.3.1. The Greedy Approach

The RPI_NHC is constructed as follows:

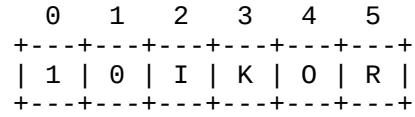


Figure 7: The RPI_NHC, Greedy Version

Depending on the RPLInstanceID and the MinHopRankIncrease, the proposed format thus squeezes the RPL information into 16 to 32 bits, which compares to 64 bits when using a Hop-by-hop option with the RPL option as specified in [RFC6553].

(This is called the "greedy" approach as it consumes 1/4 of the NHC space just for the RPI compression.)

4.3.2. The Conservative Approach

In this approach, the encoding of the RPL Packet Information takes two bytes: one byte to indicate the NH type, and then one byte to signal the compressed information.

The NH type is indicated in an extension to existing LOWPAN_NHC encodings. Section 4.2 of [RFC6553] defines LOWPAN_NHC encodings for IPv6 Extension Headers as in Figure 8:

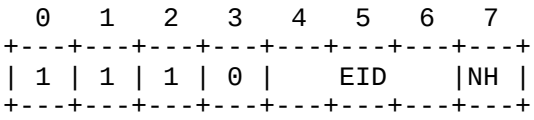


Figure 8: IPv6 Extension Header Encoding

Values 5 and 6 of the IPv6 Extension Header ID (EID) are still reserved. This specification uses EID of 5 to indicate that the next byte is a RPI_NHC. The RPI_NHC is constructed as shown in Figure 9:

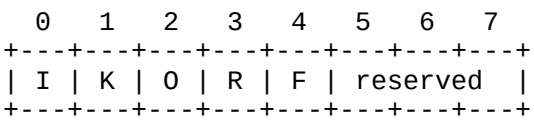


Figure 9: The RPI_NHC, Conservative Version

The bits 5 to 7 of the RPI_NHC are reserved for future use and MUST be sent as zero.

(This is called the "conservative" approach as it consumes only 1/256 of the NHC space.)

4.3.3. The Efficient Approach

4.3.3.1. The NHC escape mechanism

The NHC space of [RFC6282] is limited to 256 code points. For the case some infrequent bit combinations do not fit into the 256 code points, this specification assigns four code points:

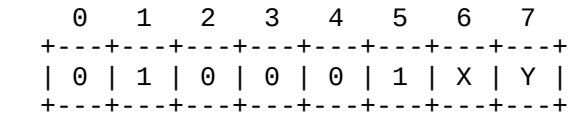


Figure 10: NHC Escape Codes

Each NHC escape code is followed by a further NHC code point. The latter MUST be a code point for which special semantics for a preceding escape code are defined, i.e., an escape code MUST NOT be used in front of an NHC code point that does not define special semantics for this escape code.

An escape code followed by another escape code supplies additional semantics; again, a sequence of such escape codes MUST NOT be used unless the final NHC code following this sequence defines the semantics for the specific sequence.

4.3.3.2. RPI_NHC Encoding

The RPI_NHC provides a compressed form for the RPI and is constructed as follows:

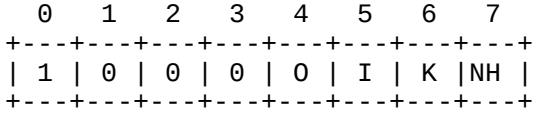


Figure 11: RPI NHC, efficient version

The R and F bits, as defined in [RFC6550], Section 11.2, are represented as follows:

If R=0 and F=0, the NHC code is used as defined above. If either is non-zero, a single escape code with X=R and Y=F is prepended in front of the NHC code. (An escape code with X=0 and Y=0 MUST NOT be used with RPI_NHC. A sequence of two or more escape codes MUST NOT be used with RPI_NHC.)

Depending on the RPLInstanceID and the MinHopRankIncrease, the proposed format thus squeezes the RPI into 16 to 40 bits, which compares to 64 bits when using a Hop-by-hop option with the RPL option as specified in [RFC6553].

(This is called the "efficient" approach as it consumes only 1/16 of the NHC space, but, depending on the frequency of set R or set F flags, is almost as efficient as the greedy approach.)

4.3.3.3. Operation

A 6lo compressor that is about to create either an RFC 6282 IPHC header [RFC6282] or a Frag1 header [RFC4944] and finds a Hop-by-Hop Options header [RFC2460] with an RPL Option [RFC6553] in it, performs the following checks:

1. Does the compression scheme apply? I.e.:

    A. is no sub-tlv present in the RPL Option?

    B. is the RPL Option the only option in the Hop-by-Hop Options header?

2. Does the additional compression for I=1 apply? I.e.: is RPLInstanceID == 0?

3. Does the additional compression for K=1 apply? I.e.: is SenderRank < 256?

4. Is both R=0 and F=0, or do we need an escape code?

If check 1 succeeds, the compressor removes the Hop-by-Hop Options header (replacing the zero-valued next header field in the IPv6 header with the value of the next header field of the Hop-by-Hop Options header), and, depending on the outcome of check 2 and 3, generates an RPI_NHC Header with I and K set from the payload information in the RPL Option. If one or both of R and F are non-zero (check 4), it precedes the first byte in the RPI_NHC header with an escape code with X=R and Y=F. It then continues generating the RFC 6282 IPHC or RFC 4944 Frag1 header, filling in the continuation of the RPL Information header as defined in Section 4.3.3.2.

A 6lo decompressor that encounters a RPL Information header reverses this process, creating a Hop-by-Hop Options header with a single RPL Option carrying the information in the RPL Information header.

5. Security Considerations

The security considerations of [RFC4944], [RFC6282], and [RFC6553] apply.

Using a compressed format as opposed to the full inline RPL option is logically equivalent and does not create an opening for a new threat when compared to [RFC6553].

6. IANA Considerations

(greedy variant:)

This document updates IANA registry for the LOWPAN_NHC defined in [RFC6282] and assigns the previously unassigned:

10IKORFN: RPL Information [RFCthis]

Capital letters in bit positions represent class-specific bit assignments. IKORF represents variables specific to RPL Information compression defined in Section 4. N indicates whether or not additional LOWPAN_NHC encodings follow, as defined in Section 4.2 of [RFC6553].

(efficient variant:)

This draft requests IANA to assign the following LOWPAN_NHC types in the "IPv6 Low Power Personal Area Network Parameters" registry:

010001XY: Escape X=0/Y=0 to X=1/Y=1 [RFCthis]

1000IOKN: RPL Information [RFCthis]

7. Acknowledgements

The author wishes to thank Laurent Toutain for suggesting this work and and Martin Turon for his constructive contributions. Ralph Droms supplied a number of helpful comments on the -00 draft of [I-D.bormann-6lo-rpl-mesh], which was superseded by the present document, turning into the "efficient approach". The discussion in the 6man and roll working groups also was helpful.

8. References

8.1. Normative References

[IEEE802154] 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", June 2011.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC6282] Hui, J. and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks", RFC 6282, September 2011.

[RFC6550] Winter, T., Thubert, P., 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, March 2012.

[RFC6552] Thubert, P., "Objective Function Zero for the Routing Protocol for Low-Power and Lossy Networks (RPL)", RFC 6552, March 2012.

[RFC6553] Hui, J. and JP. Vasseur, "The Routing Protocol for Low-Power and Lossy Networks (RPL) Option for Carrying RPL Information in Data-Plane Datagrams", RFC 6553, March 2012.

8.2. Informative References

[I-D.bormann-6lo-rpl-mesh]
Bormann, C., "NHC compression for RPL Packet Information", draft-bormann-6lo-rpl-mesh-02 (work in progress), October 2014.

[I-D.ietf-6tisch-architecture]
Thubert, P., Watteyne, T., and R. Assimiti, "An Architecture for IPv6 over the TSCH mode of IEEE 802.15.4e", draft-ietf-6tisch-architecture-03 (work in progress), July 2014.

[I-D.ietf-6tisch-tsch]
Watteyne, T., Palattella, M., and L. Grieco, "Using IEEE802.15.4e TSCH in an IoT context: Overview, Problem Statement and Goals", draft-ietf-6tisch-tsch-02 (work in progress), October 2014.

[RFC4944] Montenegro, G., Kushalnagar, N., Hui, J., and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks", RFC 4944, September 2007.

[RFC7102] Vasseur, JP., "Terms Used in Routing for Low-Power and Lossy Networks", RFC 7102, January 2014.

Authors' Addresses

Pascal Thubert (editor)
Cisco Systems
Village d'Entreprises Green Side
400, Avenue de Roumanille
Batiment T3
Biot - Sophia Antipolis 06410
FRANCE

Phone: +33 4 97 23 26 34
Email: pthubert@cisco.com

Carsten Bormann
Universitaet Bremen TZI
Postfach 330440
Bremen D-28359
Germany

Phone: +49-421-218-63921
Email: cabo@tzi.org