Internet Engineering Task Force (IETF)
Request for Comments: 8106
Obsoletes: 6106
Category: Standards Track
ISSN: 2070-1721

J. Jeong
Sungkyunkwan University
S. Park
Samsung Electronics
L. Beloeil
Orange
S. Madanapalli
NTT Data

March 2017

IPv6 Router Advertisement Options for DNS Configuration

Abstract

This document specifies IPv6 Router Advertisement (RA) options (called "DNS RA options") to allow IPv6 routers to advertise a list of DNS Recursive Server Addresses and a DNS Search List to IPv6 hosts.

This document, which obsoletes RFC 6106, defines a higher default value of the lifetime of the DNS RA options to reduce the likelihood of expiry of the options on links with a relatively high rate of packet loss.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc8106.

Table of Contents

1. Introduction

The purpose of this document is to standardize IPv6 Router Advertisement (RA) options (DNS RA options) for DNS Recursive Server Addresses used for DNS name resolution in IPv6 hosts, and also for a DNS Search List (DNSSL) of domain suffixes.

IPv6 Neighbor Discovery (ND) and IPv6 Stateless Address Autoconfiguration (SLAAC) provide ways to configure either fixed or mobile nodes with one or more IPv6 addresses, default routers, and some other parameters [RFC4861] [RFC4862].

It is infeasible to manually configure nomadic hosts each time they connect to a different network. While a one-time static configuration is possible, it is generally not desirable on general-purpose hosts such as laptops. For instance, locally defined namespaces would not be available to the host if it were to run its own recursive name server directly connected to the global DNS.

The DNS information can also be provided through DHCPv6 [RFC3315] [RFC3736] [RFC3646]. However, access to DNS is a fundamental requirement for almost all hosts, so IPv6 SLAAC cannot stand on its own as an alternative deployment model in any practical network without any support for DNS configuration.

These issues are not pressing in dual-stack networks as long as a DNS server is available on the IPv4 side, but they become more critical with the deployment of IPv6-only networks. As a result, this document defines a mechanism based on DNS RA options to allow IPv6 hosts to perform automatic DNS configuration.

1.1. Applicability Statements

RA-based DNS configuration is a useful alternative in networks where an IPv6 host’s address is autoconfigured through IPv6 SLAAC and where either (i) there is no DHCPv6 infrastructure at all or (ii) some hosts do not have a DHCPv6 client. The intention is to enable the full configuration of basic networking information for hosts without requiring DHCPv6. However, for networks that need to distribute additional information, DHCPv6 is likely to be employed. In these networks, RA-based DNS configuration may not be needed.

RA-based DNS configuration allows an IPv6 host to acquire the DNS configuration (i.e., DNS Recursive Server Addresses and the DNSSL) for the link(s) to which the host is connected. Furthermore, the host learns this DNS configuration from the same RA message that provides configuration information for the link.

The advantages and disadvantages of the RA-based approach are discussed in [RFC4339] along with other approaches, such as the DHCP and well-known anycast address approaches.

1.2. Coexistence of RA Options and DHCP Options for DNS Configuration

Two protocols exist to configure the DNS information on a host: the RA options specified in this document and the DHCPv6 options specified in [RFC3646]. They can be used together. The rules governing the decision to use stateful configuration mechanisms are specified in [RFC4861]. Hosts conforming to this specification MUST extract DNS information from RA messages, unless static DNS configuration has been specified by the user. If there is DNS information available from multiple RAs and/or from DHCP, the host MUST maintain an ordered list of this information as specified in Section 5.3.1.

2. Requirements Language

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].

3. Terminology

This document uses the terminology defined in [RFC4861] and [RFC4862]. In addition, six new terms are defined below:

- Recursive DNS Server (RDNSS): A server that provides a recursive DNS resolution service for translating domain names into IP addresses or resolving PTR records as defined in [RFC1034] and [RFC1035].

- RDNSS Option: An IPv6 RA option to deliver the RDNSS information to IPv6 hosts [RFC4861].

- DNS Search List (DNSSL): The list of DNS suffix domain names used by IPv6 hosts when they perform DNS query searches for short, unqualified domain names.

- DNSSL Option: An IPv6 RA option to deliver the DNSSL information to IPv6 hosts.

- DNS Repository: Two data structures for managing DNS configuration information in the IPv6 protocol stack, in addition to the Neighbor Cache and Destination Cache for Neighbor Discovery

[RFC4861]. The first data structure is the DNS Server List for RDNSS addresses, and the second is the DNSSL for DNS search domain names.

- Resolver Repository: Configuration repository with RDNSS addresses and a DNSSL that a DNS resolver on the host uses for DNS name resolution -- for example, the UNIX resolver file (i.e., /etc/resolv.conf) and the Windows registry.

4. Overview

This document standardizes an ND option called the "RDNSS option", which contains the addresses of RDNSSes. This document also standardizes an ND option called the "DNSSL option", which contains the DNSSL. This is to maintain parity with the DHCPv6 options and to ensure that there is necessary functionality to determine the search domains.

The existing ND message (i.e., RA) is used to carry this information. An IPv6 host can configure the IPv6 addresses of one or more RDNSSes via RA messages. Through the RDNSS and DNSSL options, along with the Prefix Information option based on the ND protocol [RFC4861] [RFC4862], an IPv6 host can perform the network configuration of its IPv6 address and the DNS information simultaneously without needing DHCPv6 for the DNS configuration. The RA options for RDNSS and DNSSL can be used on networks that support the use of ND.

This approach requires manual configuration or automatic mechanisms (e.g., DHCPv6 or vendor-proprietary configuration mechanisms) to configure the DNS information in routers sending the advertisements. The automatic configuration of RDNSS addresses and a DNSSL in routers is out of scope for this document.

5. Neighbor Discovery Extension

The IPv6 DNS configuration mechanism described in this document needs two ND options in Neighbor Discovery: (i) the RDNSS option and (ii) the DNSSL option.

5.1. Recursive DNS Server Option

The RDNSS option contains one or more IPv6 addresses of RDNSSes. All of the addresses share the same Lifetime value. If it is desirable to have different Lifetime values, multiple RDNSS options can be used. Figure 1 shows the format of the RDNSS option.

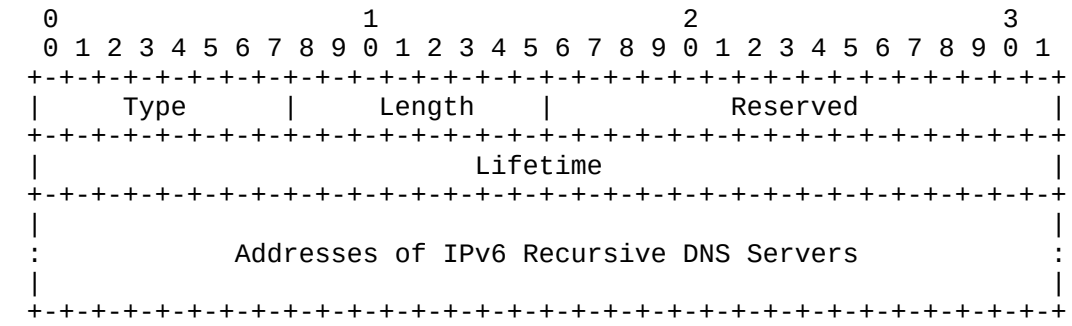

Figure 1: RDNSS Option Format

Fields:

Type 8-bit identifier of the RDNSS option type as assigned by IANA: 25

Length 8-bit unsigned integer. The length of the option (including the Type and Length fields) is in units of 8 octets. The minimum value is 3 if one IPv6 address is contained in the option. Every additional RDNSS address increases the length by 2. The Length field is used by the receiver to determine the number of IPv6 addresses in the option.

Lifetime 32-bit unsigned integer. The maximum time in seconds (relative to the time the packet is received) over which these RDNSS addresses MAY be used for name resolution. The value of Lifetime SHOULD by default be at least 3 * MaxRtrAdvInterval, where MaxRtrAdvInterval is the maximum RA interval as defined in [RFC4861]. A value of all one bits (0xffffffff) represents infinity. A value of zero means that the RDNSS addresses MUST no longer be used.

Addresses of IPv6 Recursive DNS Servers
: One or more 128-bit IPv6 addresses of the RDNSSes. The number of addresses is determined by the Length field. That is, the number of addresses is equal to (Length - 1) / 2.

Note: The addresses for RDNSSes in the RDNSS option MAY be link-local addresses. Such link-local addresses SHOULD be registered in the Resolver Repository along with the corresponding link zone indices of the links that receive the RDNSS option(s) for them. The link-local addresses MAY be represented in the Resolver Repository with their link zone indices in the textual format for scoped addresses as described in [RFC4007]. When a resolver sends a DNS query message to an RDNSS identified by a link-local address, it MUST use the corresponding link.

The rationale of the default value of the Lifetime field is as follows. The Router Lifetime field, set by AdvDefaultLifetime, has the default of 3 * MaxRtrAdvInterval as specified in [RFC4861], so such a default or a larger default can allow for the reliability of DNS options even under the loss of RAs on links with a relatively high rate of packet loss. Note that the ratio of AdvDefaultLifetime to MaxRtrAdvInterval is the number of unsolicited multicast RAs sent by the router. Since the DNS option entries can survive for at most three consecutive losses of RAs containing DNS options, the default value of the Lifetime lets the DNS option entries be resilient to packet-loss environments.

5.2. DNS Search List Option

The DNSSL option contains one or more domain names of DNS suffixes. All of the domain names share the same Lifetime value. If it is desirable to have different Lifetime values, multiple DNSSL options can be used. Figure 2 shows the format of the DNSSL option.

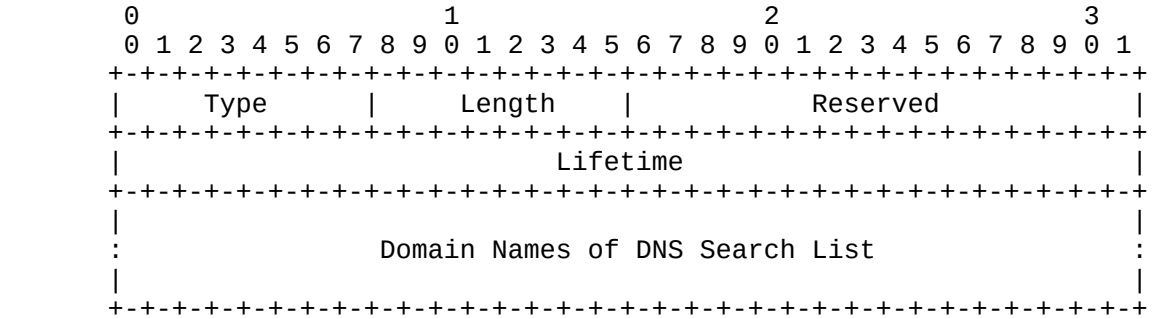

Figure 2: DNSSL Option Format

Fields:

Type 8-bit identifier of the DNSSL option type as assigned by IANA: 31

Length 8-bit unsigned integer. The length of the option (including the Type and Length fields) is in units of 8 octets. The minimum value is 2 if at least one domain name is contained in the option. The Length field is set to a multiple of 8 octets to accommodate all the domain names in the "Domain Names of DNS Search List" field.

Lifetime 32-bit unsigned integer. The maximum time in seconds (relative to the time the packet is received) over which these DNSSL domain names MAY be used for name resolution. The Lifetime value has the same semantics as the semantics for the RDNSS option. That is, Lifetime SHOULD by default be at least 3 * MaxRtrAdvInterval. A value of all one bits (0xffffffff) represents infinity. A value of zero means that the DNSSL domain names MUST no longer be used.

Domain Names of DNS Search List
One or more domain names of the DNSSL that MUST be encoded as described in Section 3.1 of [RFC1035]. With this technique, each domain name is represented as a sequence of labels ending in a zero octet, defined as a domain name representation. For more than one domain name, the corresponding domain name representations are concatenated as they are. Note that for the simple decoding, the domain names MUST NOT be encoded in the compressed form described in Section 4.1.4 of [RFC1035]. Because the size of this field MUST be a multiple of 8 octets, for the minimum multiple including the domain name representations, the remaining octets other than the encoding parts of the domain name representations MUST be padded with zeros.

5.3. DNS Configuration Procedure

The procedure for DNS configuration through the RDNSS and DNSSL options is the same as it is with any other ND option [RFC4861].

5.3.1. Procedure in IPv6 Hosts

When an IPv6 host receives DNS options (i.e., RDNSS and DNSSL options) through RA messages, it processes the options as follows:

- The validity of DNS options is checked with the Length field; that is, the value of the Length field in the RDNSS option is greater than or equal to the minimum value (3) and satisfies the requirement that (Length - 1) % 2 == 0. The value of the Length field in the DNSSL option is greater than or equal to the minimum value (2). Also, the validity of the RDNSS option is checked with the "Addresses of IPv6 Recursive DNS Servers" field; that is, the addresses should be unicast addresses.

- If the DNS options are valid, the host SHOULD copy the values of the options into the DNS Repository and the Resolver Repository in order. Otherwise, the host MUST discard the options. Refer to Section 6 for the detailed procedure.

In the case where the DNS information of RDNSS and DNSSL can be obtained from multiple sources, such as RAs and DHCP, the IPv6 host SHOULD keep some DNS options from all sources. Unless explicitly specified for the discovery mechanism, the exact number of addresses and domain names to keep is a matter of local policy and implementation choice as a local configuration option. However, in the case of multiple sources, the ability to store a total of at least three RDNSS addresses (or DNSSL domain names) from the multiple sources is RECOMMENDED. The DNS options from RAs and DHCP SHOULD be stored in the DNS Repository and Resolver Repository so that information from DHCP appears there first and therefore takes precedence. Thus, the DNS information from DHCP takes precedence over that from RAs for DNS queries. On the other hand, for DNS options announced by RAs, if some RAs use the Secure Neighbor Discovery (SEND) protocol [RFC3971] for RA security, they MUST be preferred over those that do not use SEND. Also, DNS options announced by RAs via SEND MUST be preferred over those announced by unauthenticated DHCP [RFC3118]. Refer to Section 7 for a detailed discussion of SEND for DNS RA options.

5.3.2. Warnings for DNS Options Configuration

There are two warnings for DNS options configuration: (i) warning for multiple sources of DNS options and (ii) warning for multiple network interfaces. First, in the case of multiple sources for DNS options (e.g., RAs and DHCP), an IPv6 host can configure its IP addresses from these sources. In this case, it is not possible to control how the host uses DNS information and what source addresses it uses to send DNS queries. As a result, configurations where different

information is provided by different mechanisms for autoconfiguration may lead to problems. Therefore, the network administrator needs to carefully configure different DNS options in the multiple mechanisms for autoconfiguration in order to minimize the impact of such problems [DHCPv6-SLAAC].

Second, if different DNS information is provided on different network interfaces, this can lead to inconsistent behavior. The IETF worked on solving this problem for both DNS and other information obtained from multiple interfaces [RFC6418] [RFC6419] and standardized a DHCP-based solution for RDNSS selection for multi-interfaced nodes as described in [RFC6731].

6. Implementation Considerations

The implementation considerations in this document include the following three: (i) DNS repository management, (ii) synchronization between the DNS Server List and the Resolver Repository, and (iii) synchronization between the DNSSL and the Resolver Repository.

Note: The implementations that are updated according to this document will still interoperate with the existing implementations according to [RFC6106]. This is because the main change in this document is the increase of the default Lifetime of DNS options, considering lossy links.

6.1. DNS Repository Management

For DNS repository management, the following two data structures SHOULD be synchronized with the Resolver Repository: (i) the DNS Server List, which keeps the list of RDNSS addresses and (ii) the DNSSL, which keeps the list of DNS search domain names. Each entry in these two lists consists of a pair of an RDNSS address (or DNSSL domain name) and Expiration-time as follows:

- RDNSS address for DNS Server List: IPv6 address of the RDNSS that is available for recursive DNS resolution service in the network advertising the RDNSS option.

- DNSSL domain name for DNSSL: DNS suffix domain name that is used to perform DNS query searches for short, unqualified domain names.

- Expiration-time for DNS Server List or DNSSL: The time when this entry becomes invalid. Expiration-time is set to the value of the Lifetime field of the RDNSS option or DNSSL option plus the current time. Whenever a new RDNSS option with the same address (or DNSSL option with the same domain name) is received on the same interface as a previous RDNSS option (or DNSSL option), this

field is updated to have a new Expiration-time. When the current time becomes larger than Expiration-time, this entry is regarded as expired, so it should not be used any more. Note that the DNS information for the RDNSS and DNSSL options need not be dropped if the expiry of the RA router lifetime happens. This is because these options have their own lifetime values.

6.2. Synchronization between DNS Server List and Resolver Repository

When an IPv6 host receives the information of multiple RDNSS addresses within a network (e.g., campus network and company network) through an RA message with RDNSS option(s), it stores the RDNSS addresses (in order) in both the DNS Server List and the Resolver Repository. The processing of the RDNSS consists of (i) the processing of RDNSS option(s) included in an RA message and (ii) the handling of expired RDNSSes. The processing of RDNSS option(s) is as follows:

- Step (a): Receive and parse the RDNSS option(s). For the RDNSS addresses in each RDNSS option, perform Steps (b) through (d).

- Step (b): For each RDNSS address, check the following: If the RDNSS address already exists in the DNS Server List and the RDNSS option's Lifetime field is set to zero, delete the corresponding RDNSS entry from both the DNS Server List and the Resolver Repository in order to prevent the RDNSS address from being used any more for certain reasons in network management, e.g., the termination of the RDNSS or a renumbering scenario. That is, the RDNSS can resign from its DNS service because the machine running the RDNSS is out of service intentionally or unintentionally. Also, in the renumbering scenario, the RDNSS's IPv6 address will be changed, so the previous RDNSS address should not be used any more. The processing of this RDNSS address is finished here. Otherwise, go to Step (c).

- Step (c): For each RDNSS address, if it already exists in the DNS Server List and the RDNSS option's Lifetime field is not set to zero, then just update the value of the Expiration-time field according to the procedure specified in the third bullet of Section 6.1. Otherwise, go to Step (d).

- Step (d): For each RDNSS address, if it does not exist in the DNS Server List, register the RDNSS address and Lifetime with the DNS Server List and then insert the RDNSS address as the first one in the Resolver Repository. In the case where the data structure for the DNS Server List is full of RDNSS entries (that is, has more RDNSSes than the sufficient number discussed in Section 5.3.1), delete from the DNS Server List the entry with the shortest

Expiration-time (i.e., the entry that will expire first). The corresponding RDNSS address is also deleted from the Resolver Repository. For the ordering of RDNSS addresses in an RDNSS option, position the first RDNSS address in the RDNSS option as the first one in the Resolver Repository, the second RDNSS address in the option as the second one in the repository, and so on. This ordering allows the RDNSS addresses in the RDNSS option to be preferred according to their order in the RDNSS option for DNS name resolution. The processing of these RDNSS addresses is finished here.

The handling of expired RDNSSes is as follows: Whenever an entry expires in the DNS Server List, the expired entry is deleted from the DNS Server List, and also the RDNSS address corresponding to the entry is deleted from the Resolver Repository.

6.3. Synchronization between DNS Search List and Resolver Repository

When an IPv6 host receives the information of multiple DNSSL domain names within a network through an RA message with DNSSL option(s), it stores the DNSSL domain names (in order) in both the DNSSL and the Resolver Repository. The processing of the DNSSL consists of (i) the processing of DNSSL option(s) included in an RA message and (ii) the handling of expired DNSSLs. The processing of DNSSL option(s) is the same as the processing of RDNSS option(s) as described in Section 6.2.

7. Security Considerations

In this section, we analyze security threats related to DNS options and then make recommendations to cope with such security threats.

7.1. Security Threats

For the RDNSS option, an attacker could send an RA with a fraudulent RDNSS address, misleading IPv6 hosts into contacting an unintended DNS server for DNS name resolution. Also, for the DNSSL option, an attacker can let IPv6 hosts resolve a hostname without a DNS suffix into an unintended host’s IP address with a fraudulent DNSSL. These attacks are similar to ND attacks specified in [RFC4861] that use Redirect or Neighbor Advertisement messages to redirect traffic to individual addresses of malicious parties.

However, the security of these RA options for DNS configuration does not affect ND protocol security [RFC4861]. This is because learning DNS information via the RA options cannot be worse than learning bad router information via the RA options. Therefore, the vulnerability of ND is not worse and is a subset of the attacks that any node attached to a LAN can do.

7.2. Recommendations

The Secure Neighbor Discovery (SEND) protocol [RFC3971] is designed as a security mechanism for ND. In this case, ND can use SEND to allow all the ND options, including the RDNSS and DNSSL options, to be automatically signed with digital signatures.

It is common for network devices such as switches to include mechanisms to block unauthorized ports from running a DHCPv6 server to provide protection from rogue DHCPv6 servers [RFC7610]. That means that an attacker on other ports cannot insert bogus DNS servers using DHCPv6. The corresponding technique for network devices is RECOMMENDED to block rogue RA messages that include the RDNSS and DNSSL options from unauthorized nodes [RFC6104] [RFC6105].

An attacker may provide a bogus DNSSL option in order to cause the victim to send DNS queries to a specific DNS server when the victim queries non-FQDNs (fully qualified domain names). For this attack, the DNS resolver in IPv6 hosts can mitigate the vulnerability with the recommendations mentioned in [RFC1535], [RFC1536], and [RFC3646].

8. IANA Considerations

The RDNSS option defined in this document uses the IPv6 Neighbor Discovery Option type assigned by IANA as follows:

Option Name	Type
Recursive DNS Server Option	25

The DNSSL option defined in this document uses the IPv6 Neighbor Discovery Option type assigned by IANA as follows:

Option Name	Type
DNS Search List Option	31

These options are registered in the "IPv6 Neighbor Discovery Option Formats" registry [ICMPv6].

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, DOI 10.17487/RFC4861, September 2007, <http://www.rfc-editor.org/info/rfc4861>.

[RFC4862] Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless Address Autoconfiguration", RFC 4862, DOI 10.17487/RFC4862, September 2007, <http://www.rfc-editor.org/info/rfc4862>.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <http://www.rfc-editor.org/info/rfc1035>.

[RFC4007] Deering, S., Haberman, B., Jinmei, T., Nordmark, E., and B. Zill, "IPv6 Scoped Address Architecture", RFC 4007, DOI 10.17487/RFC4007, March 2005, <http://www.rfc-editor.org/info/rfc4007>.

9.2. Informative References

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <http://www.rfc-editor.org/info/rfc1034>.

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

[RFC3736] Droms, R., "Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6", RFC 3736, DOI 10.17487/RFC3736, April 2004, <http://www.rfc-editor.org/info/rfc3736>.

[RFC3646] Droms, R., Ed., "DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3646, DOI 10.17487/RFC3646, December 2003, <http://www.rfc-editor.org/info/rfc3646>.

[RFC6106] Jeong, J., Park, S., Beloeil, L., and S. Madanapalli, "IPv6 Router Advertisement Options for DNS Configuration", RFC 6106, DOI 10.17487/RFC6106, November 2010, <http://www.rfc-editor.org/info/rfc6106>.

[RFC4339] Jeong, J., Ed., "IPv6 Host Configuration of DNS Server Information Approaches", RFC 4339, DOI 10.17487/RFC4339, February 2006, <http://www.rfc-editor.org/info/rfc4339>.

[RFC3971] Arkko, J., Ed., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, DOI 10.17487/RFC3971, March 2005, <http://www.rfc-editor.org/info/rfc3971>.

[RFC3118] Droms, R., Ed., and W. Arbaugh, Ed., "Authentication for DHCP Messages", RFC 3118, DOI 10.17487/RFC3118, June 2001, <http://www.rfc-editor.org/info/rfc3118>.

[RFC6104] Chown, T. and S. Venaas, "Rogue IPv6 Router Advertisement Problem Statement", RFC 6104, DOI 10.17487/RFC6104, February 2011, <http://www.rfc-editor.org/info/rfc6104>.

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

[RFC7610] Gont, F., Liu, W., and G. Van de Velde, "DHCPv6-Shield: Protecting against Rogue DHCPv6 Servers", BCP 199, RFC 7610, DOI 10.17487/RFC7610, August 2015, <http://www.rfc-editor.org/info/rfc7610>.

[RFC1535] Gavron, E., "A Security Problem and Proposed Correction With Widely Deployed DNS Software", RFC 1535, DOI 10.17487/RFC1535, October 1993, <http://www.rfc-editor.org/info/rfc1535>.

[RFC1536] Kumar, A., Postel, J., Neuman, C., Danzig, P., and S. Miller, "Common DNS Implementation Errors and Suggested Fixes", RFC 1536, DOI 10.17487/RFC1536, October 1993, <http://www.rfc-editor.org/info/rfc1536>.

[DHCPv6-SLAAC] Liu, B., Jiang, S., Gong, X., Wang, W., and E. Rey, "DHCPv6/SLAAC Interaction Problems on Address and DNS Configuration", Work in Progress, draft-ietf-v6ops-dhcpv6-slaac-problem-07, August 2016.

[RFC6418] Blanchet, M. and P. Seite, "Multiple Interfaces and Provisioning Domains Problem Statement", RFC 6418, DOI 10.17487/RFC6418, November 2011, <http://www.rfc-editor.org/info/rfc6418>.

[RFC6419] Wasserman, M. and P. Seite, "Current Practices for Multiple-Interface Hosts", RFC 6419, DOI 10.17487/RFC6419, November 2011, <http://www.rfc-editor.org/info/rfc6419>.

[RFC6731] Savolainen, T., Kato, J., and T. Lemon, "Improved Recursive DNS Server Selection for Multi-Interfaced Nodes", RFC 6731, DOI 10.17487/RFC6731, December 2012, <http://www.rfc-editor.org/info/rfc6731>.

[ICMPv6] IANA, "Internet Control Message Protocol version 6 (ICMPv6) Parameters", <http://www.iana.org/assignments/icmpv6-parameters/>.

Appendix A. Changes from RFC 6106

The following changes were made from RFC 6106 ("IPv6 Router Advertisement Options for DNS Configuration"):

- This document allows a higher default value of the lifetime of the DNS RA options than RFC 6106 in order to avoid the frequent expiry of the options on links with a relatively high rate of packet loss; at the same time, this document also makes additional clarifications. The lifetime's lower bound of 2 * MaxRtrAdvInterval was shown to lead to the expiry of these options on links with a relatively high rate of packet loss. To avoid this problem, this revision relaxes the lower bound and sets a higher default value of 3 * MaxRtrAdvInterval.

- The text regarding the generation of a Router Solicitation message to ensure that the RDNSS information is fresh before the expiry of the RDNSS option is removed in order to prevent multicast traffic on the link from increasing.

- The addresses for RDNSSes in the RDNSS option can be not only global addresses but also link-local addresses. The link-local addresses for RDNSSes should be registered in the Resolver Repository along with the corresponding link zone indices.

- RFC 6106 recommended that the number of RDNSS addresses that should be learned and maintained through the RDNSS RA option should be limited to three. This document removes that recommendation; thus, the number of RDNSS addresses to maintain is determined by an implementer's local policy.

- RFC 6106 recommended that the number of DNS search domains that should be learned and maintained through the DNSSL RA option should be limited to three. This document removes that recommendation; thus, when the set of unique DNSSL values are not equivalent, none of them may be ignored for hostname lookups according to an implementer's local policy.

- The guidance of the specific implementation for the synchronization of the DNS Repository and Resolver Repository in the kernel space and user space is removed.

- The key words "SHOULD" and "RECOMMENDED" (RFC 2119) are removed in the recommendation of using SEND as a security mechanism for ND. Instead of using these key words, SEND is specified as only a possible security mechanism for ND.

Acknowledgements

This document has greatly benefited from inputs by Robert Hinden, Pekka Savola, Iljitsch van Beijnum, Brian Haberman, Tim Chown, Erik Nordmark, Dan Wing, Jari Arkko, Ben Campbell, Vincent Roca, Tony Cheneau, Fernando Gont, Jen Linkova, Ole Troan, Mark Smith, Tatuya Jinmei, Lorenzo Colitti, Tore Anderson, David Farmer, Bing Liu, and Tassos Chatzithomaoglou. The authors sincerely appreciate their contributions.

This document was supported by an Institute for Information & communications Technology Promotion (IITP) grant funded by the Korean government (MSIP) [10041244, Smart TV 2.0 Software Platform].

Authors' Addresses

Jaehoon Paul Jeong
Department of Software
Sungkyunkwan University
2066 Seobu-Ro, Jangan-Gu
Suwon, Gyeonggi-Do 16419
Republic of Korea

Phone: +82 31 299 4957
Fax: +82 31 290 7996
Email: pauljeong@skku.edu
URI: http://iotlab.skku.edu/people-jaehoon-jeong.php

Soohong Daniel Park
Software R&D Center
Samsung Electronics
Seoul R&D Campus D-Tower, 56, Seongchon-Gil, Seocho-Gu
Seoul 06765
Republic of Korea

Email: soohong.park@samsung.com

Luc Beloeil
Orange
5 rue Maurice Sibille
BP 44211
44042 Nantes Cedex 1
France

Phone: +33 2 28 56 11 84
Email: luc.beloeil@orange.com

Syam Madanapalli
NTT Data
#H304, Shriram Samruddhi, Thubarahalli
Bangalore 560066
India

Phone: +91 959 175 7926
Email: smadanapalli@gmail.com