DHC Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 15, 2015

S. Jiang
Huawei Technologies Co., Ltd
G. Chen
China Mobile
S. Krishnan
Ericsson
R. Asati
Cisco Systems, Inc.

September 11, 2014

# Registering Self-generated IPv6 Addresses in DNS using DHCPv6
# draft-ietf-dhc-addr-registration-07

## Abstract

In networks that are centrally managed, self-generated addresses cause some traceability issues due to their decentralized nature. One of the most important issues in this regard is the inability to register such addresses in DNS. This document defines a mechanism to register self-generated and statically configured addresses in DNS through a DHCPv6 server.

## 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 March 15, 2015.

Table of Contents



## 1. Introduction

In several common network scenarios, IPv6 addresses are self-generated by the end-hosts by appending a self-generated interface identifier to a network-specified prefix. Examples of self-generated addresses include those created using IPv6 Stateless Address Configuration [RFC4862] , temporary addresses [RFC4941] and Cryptographically Generated Addresses (CGA) [RFC3972] etc. In several tightly controlled networks, hosts with self-generated addresses may face some limitations. One such limitation is related to the inability of nodes with self-generated addresses to register their IPv6-address-to-FQDN bindings in DNS. This is related to the fact that, in such networks, only certain nodes (e.g. The DHCPv6 server) are allowed to update these bindings in order to prevent end-hosts from registering arbitrary addresses for their FQDNs or associating their addresses with arbitrary domain names. The administrators may not want to distribute the address of authoritative name-server. Also, there is no way to propagate the address of authoritative name server by any protocols. It is preferred that the address registration server, which is under the same management with the authoritative name-server, to know the address of the authoritative name-server and make registration requests on behalf of clients. It is preferred by administrators to

establish and manage one trust relationship between a single DHCPv6 (address registration) server and the DNS authoritative name-server, rather than to distribute and manage trust relationships between many clients and the DNS authoritative name-server.

For nodes that obtain their addresses through DHCPv6, a solution has been specified in [RFC4704]. The solution works by including a Client FQDN option in the SOLICIT, REQUEST, RENEW or REBIND messages during the process of acquiring an address through DHCPv6. This document provides an analogous mechanism to register self-generated addresses in DNS.

A new ADDR-REGISTRATION-REQUEST DHCPv6 message type is defined to initiate the address registration request, and two new Status codes are defined to indicate registration errors on the server side.

## 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].

Certificate In this document, the term "Certificate" is all referred to public key certificate.

## 3. Solution Overview

After successfully assigning a self-generated IPv6 address on one of its interfaces, an end-host implementing this specification SHOULD send an ADDR-REGISTRATION-REQUEST message to a DHCPv6 address registration server. After receiving the address registration request, the DHCPv6 server registers the IPv6 address to FQDN binding towards a configured DNS server. An acknowledgement MUST be sent back to the end host to indicate whether or not the registration operation succeeded.

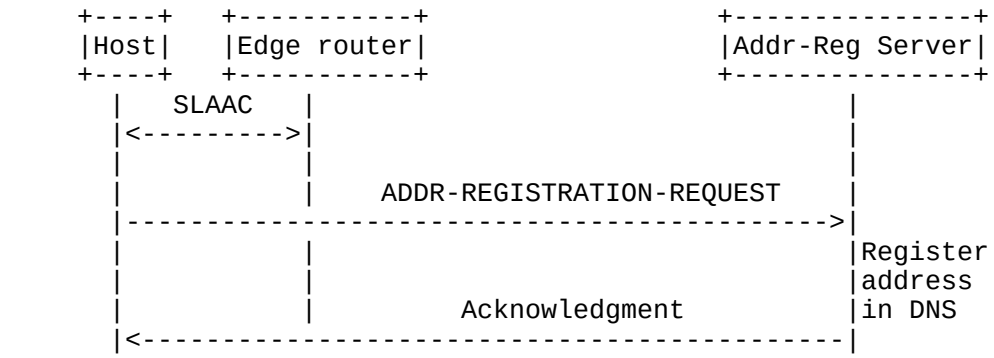


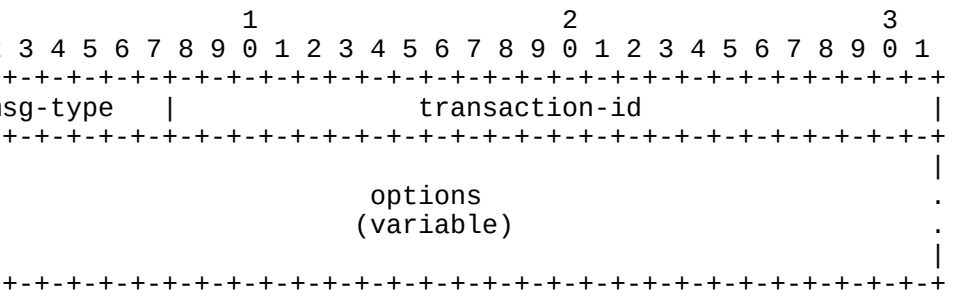

Figure 1: Address Registration Procedure

Furthermore, the registration server MAY apply certain filter/accept criteria for the address registration requests, particularly for the client chosen domain names.

It is RECOMMENDED to only set up one address registration server within an administration domain, although there may be multiple DHCPv6 servers. While using multiple address registration servers does potentially increase the load on DNS, because of how [RFC4703] and [RFC4704] work, this should NOT be an issue - the servers should work correctly in updating DNS (either adding or removing the entries). The broken part with multiple servers is the 'extension' of the registration. If there are two address registration servers and both receive the initial registration and (correctly) update DNS, the problem comes when the client extends this but one of the servers does not receive this extension. Then, the server that missed the extension removes the entry prematurely (i.e., when it expired originally).

## 4. DHCPv6 ADDR-REGISTRATION-REQUEST Message

The DHCPv6 client sends an ADDR-REGISTRATION-REQUEST message to a server to request an address to be registered in the DNS. The format of the ADDR-REGISTRATION-REQUEST message is described as follows:




msg-type — Identifies the DHCPv6 message type; Set to ADDR-REGISTRATION-REQUEST (TBA1).

transaction-id — The transaction ID for this message exchange.

options — Options carried in this message.

DHCPv6 ADDR-REGISTRATION-REQUEST message

The ADDR-REGISTRATION-REQUEST message MUST NOT contain server-identifier option and MUST contain the IA Address option and the DHCPv6 FQDN option [RFC4704]. The ADDR-REGISTRATION-REQUEST message is dedicated for clients to initiate an address registration request toward an address registration server. Consequently, clients MUST NOT put any Option Request Option(s) in the ADDR-REGISTRATION-REQUEST message.

Clients MUST discard any received ADDR-REGISTRATION-REQUEST messages.

Servers MUST discard any ADDR-REGISTRATION-REQUEST messages that meet any of the following conditions:

- the message does not include a Client Identifier option;
- the message includes a Server Identifier option;
- the message does not include at least one IA Address option;
- the message does not include FQDN option (or include multiple FQDN options);
- the message includes an Option Request Option.

## 5. DHCPv6 Address Registration Procedure

The DHCPv6 protocol is used as the address registration protocol when a DHCPv6 server performs the role of an address registration server. The DHCPv6 IA Address option [RFC3315] and the DHCPv6 FQDN option

[RFC4704] are adopted in order to fulfill the address registration interactions.

### 5.1. DHCPv6 Address Registration Request

The end-host sends a DHCPv6 ADDR-REGISTRATION-REQUEST message to the address registration server to the All_DHCP_Relay_Agents_and_Servers multicast address (ff02::1:2).

The end-host MUST include a Client Identifier option in the ADDR-REGISTRATION-REQUEST message to identify itself to the server. The DHCPv6 ADDR-REGISTRATION-REQUEST message MUST contain at least one IA Address option and exactly one FQDN option. The valid-lifetime field of the IA Address option MUST be set to the period for which the client would like to register the binding in DNS.

After receiving this ADDR-REGISTRATION-REQUEST message, the address registration server MUST register the binding between the provided FQDN and address(es) in DNS. If the DHCPv6 server does not support address registration function, it MUST sliently drop the message.

### 5.2. Registration Expiry and Refresh

For every successful binding registration, the address registration server MUST record the IPv6-address-to-FQDN bindings and associated valid-lifetimes in its storage.

The address registration client MUST refresh the registration before it expires (i.e. before the valid-lifetime of the IA address elapses) by sending a new ADDR-REGISTRATION-REQUEST to the address registration server. If the address registration server does not receive such a refresh after the valid-lifetime has passed, it SHOULD remove the IPv6-address-to-FQDN bindings in DNS, also the local record.

It is RECOMMENDED that clients initiate a refresh at about 85% of the valid-lifetime. Because RAs may periodically ’reset’ the valid-lifetime, the refresh timer MUST be independently maintained from the address valid-lifetime. Clients SHOULD set a refresh timer to 85% of the valid-lifetime when they complete a registration operation and only update this timer if 85% of any updated valid-lifetime would be sooner than the timer.

### 5.3. Acknowledging Registration and Retransmission

After an address registration server accepts an address registration request, it MUST send a Reply message as the response to the client. The acceptence reply only means that the server has taken

responsiblity to registry for the client. It may not have actually completed the update yet. The server is responsible to register all the addresses in DNS. The server generates a Reply message and includes a Status Code option with value Success, a Server Identifier option with the server's DUID, and a Client Identifier option with the client's DUID.

If there is no reply received within some interval, the client SHOULD retransmits the message according to section 14 of [RFC3315], using the following parameters:

- IRT ADDR_REG_TIMEOUT
- MRT ADDR_REG_MAX_RT
- MRC ADDR_REG_MAX_RC
- MRD 0

The below presents a table of values used to describe the message transmission behavior of clients and servers:

| Parameter | Default | Description |
|---|---|---|
| ADDR_REG_TIMEOUT | 1 secs | Initial Addr Registration Request timeout |
| ADDR_REG_MAX_RT | 60 secs | Max Addr Registration Request timeout value |
| ADDR_REG_MAX_RC | 5 | Max Request retry attempts |

For each IA Address option in the ADDR-REGISTRATION-REQUEST message for which the server does not accept its associated registration request, the server adds an IA Address option with the associated IPv6 address, and includes a Status Code option with the value RegistrationDenied (TBA2) in the IA Address option. No other options are included in the IA Address option.

Upon receiving a RegistrationDenied error status code, the client MAY also resend the message following normal retransmission routines defined in [RFC3315] with above parameters. The client MUST wait out the retransmission time before retrying.

## 6. Security Considerations

An attacker may attempt to register large number of addresses in quick succession in order to overwhelm the address registration server. These attacks may be prevented generic DHCPv6 protection by using the AUTH option [RFC3315] or Secure DHCPv6 [I-D.ietf-dhc-sedhcpv6].

## 7. IANA Considerations

This document defines a new DHCPv6 message, the ADDR-REGISTRATION-REQUEST message (TBA1) described in Section 4, that requires an allocation out of the registry of Message Types defined at http://www.iana.org/assignments/dhcpv6-parameters/

| Value | Description | Reference |
|---|---|---|
| TBA1 | ADDR-REGISTRATION-REQUEST | this document |

This document defines a new DHCPv6 Status code, the RegistrationDenied (TBA2) described in Section 5, that requires an allocation out of the registry of Status Codes defined at http://www.iana.org/assignments/dhcpv6-parameters/

| Code | Name | Reference |
|---|---|---|
| TBA2 | RegistrationDenied | this document |

## 8. Acknowledgements

The authors would like to thank Ralph Droms, Ted Lemon, Bernie Volz, Sten Carlsen, Erik Kline, Lorenzo Colitti, Joel Jaeggli, Sten Carlsen, Mark Smith, Marcin Siodelski, Darpan Malhotra, Tomek Mrugalski, Jinmei Tatuya and other members of dhc and v6ops working groups for their valuable comments.

## 9. References

### 9.1. Normative References

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

[RFC2136] Vixie, P., Thomson, S., Rekhter, Y., and J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", RFC 2136, April 1997.

[RFC3315] Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C., and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3315, July 2003.

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

[RFC4703] Stapp, M. and B. Volz, "Resolution of Fully Qualified Domain Name (FQDN) Conflicts among Dynamic Host Configuration Protocol (DHCP) Clients", RFC 4703, October 2006.

[RFC4704] Volz, B., "The Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Client Fully Qualified Domain Name (FQDN) Option", RFC 4704, October 2006.

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

[RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 4941, September 2007.

## 9.2. Informative References

[I-D.ietf-dhc-sedhcpv6]
Jiang, S., Shen, S., Zhang, D., and T. Jinmei, "Secure DHCPv6 with Public Key", draft-ietf-dhc-sedhcpv6-03 (work in progress), June 2014.

## Authors' Addresses

Sheng Jiang
Huawei Technologies Co., Ltd
Q14, Huawei Campus
No.156 Beiqing Road
Hai-Dian District, Beijing 100095
P.R. China

Email: jiangsheng@huawei.com

Gang Chen
China Mobile
53A, Xibianmennei Ave., Xuanwu District, Beijing
P.R. China

Phone: 86-13910710674
Email: phdgang@gmail.com

Suresh Krishnan
Ericsson
8400 Decarie Blvd.
Town of Mount Royal, QC
Canada

Phone: +1 514 345 7900 x42871
Email: suresh.krishnan@ericsson.com

Rajiv Asati
Cisco Systems, Inc.
7025 Kit Creek road
Research Triangle Park, NC 27709-4987
USA

Email: rajiva@cisco.com