Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: September 20, 2018
G. Ren
L. He
Y. Liu
Tsinghua University
March 19, 2018

# Problem Statement of Multi-requirement Extensions for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)

draft-ren-dhc-problem-statement-of-mredhcpv6-00

## Abstract

The manageability, security, privacy protection, and traceability of networks can be supported by extending DHCPv6 protocol. This document analyzes the current extension practices and typical DHCP server software on extensions, defines a DHCP general model, lists the unresolved problem, and provides the possible directions to solve the problems.

## Status of This Memo



## Copyright Notice

Table of Contents



## 1. Introduction

The IP address plays a significant role in the communication of the current Internet. Also, IP address generation is closely related to the manageability, security, privacy protection, and traceability of the networks. Dynamic Host Configuration Protocol for IPv6 (DHCPv6) [RFC3315] is an important network protocol that can be used to dynamically provide IPv6 addresses and other network configuration parameters to IPv6 hosts. Actually, DHCPv6 continues to be extended and improved through being added new options and defined new protocols or message processing mechanisms.

Even if DHCPv6 provides more and more comprehensive functionality and many DHCPv6 server software have provided extension interfaces to allow users to alter and customize the way how they handle and respond to DHCPv6 messages, a general insight of how to solve the extension problem effectively is lack. We should figure out where and how to conduct extensions in the DHCPv6. Therefore, a detailed analysis is required to clarify the problems, design principles, and extract and unify the design specifications to help better solve the extension problems.

In summary, multiple extensions can be conducted on DHCPv6 to support the user's self-defined functionality. As DHCPv6 is such an important and useful protocol related to IPv6 addresses generation, it can provide more extended and flexible functionality to meet users' requirements. According to well-designed principles, extended interfaces can be defined to support more self-defined multi-requirement extensions without sacrificing the stability of DHCPv6.

Some people would suggest the user modify the open-source DHCP servers to solve their own problem. However, a great amount of time will be taken to understand the open source DHCP server code, not to say the consuming time debugging the bugs, failures or system crash caused by modifying the complicated modules. Another problem is that as the open source software evolves, the source code of the server software may change (new functionality or fixing bugs). Users may need to re-write their codes once the new version of open-source server software comes out [kea_dhcp_hook_developers_guide] . Hence, the multi-requirement extensions for DHCPv6 to solve users' specific problems are very necessary and significant.

This document describes the current extension practices and typical DHCP server software on extensions and provides a problem statement by defining a DHCP general model, analyzing the unresolved multi-requirement extension problems, and providing possible directions for new work that could solve these problems.

## 2. Terminology

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

Familiarity with DHCPv6 and its terminology, as defined in [RFC3315], is assumed.

## 3. Current Extension Practices

### 3.1. Standardized and Non-standardized DHCPv6 Extension Cases

Many documents attempt to extend the DHCPv6. They can be classified into three categories.

Extended options    Most extensions for DHCPv6 are implemented in this way, e.g., DNS configurations [RFC3646], NIS configurations [RFC3898], SNTP configurations [RFC4075], information refresh time configurations [RFC4242], remote-id [RFC4649], FQDN configurations [RFC4704], relay agent echo request [RFC4994], network boot [RFC5970], client link-layer address [RFC6939]. New-defined options carry specific parameters in the DHCPv6 messages, which helps DHCPv6 clients or servers know the detailed situation with each other.

Extended messages — Some documents define new protocols to achieve specific purposes, e.g., secure DHCPv6 [draft-ietf-dhc-sedhcpv6-21], active leasequery [RFC7653], GAGMS [GAGMS]. These protocols often requires defining new messages and new options.

Extended entities — Some documents introduce extra entities into the communications of DHCPv6 to achieve specific purpose, e.g., authentication [RFC7037].

## 3.2. Current DHCPv6 Server Software Cases

Many commercial and open source DHCP server software exist, including Cisco Prime Network Registrar [CPNR], Microsoft DHCP [Microsoft_DHCP], VitalQIP [VitalQIP], Nominum DHCP [Nominum_DHCP], ISC DHCP [ISC_DHCP], Kea DHCP [Kea_DHCP], FreeRADIUS DHCP [FreeRADIUS_DHCP], WIDE DHCPv6 [WIDE_DHCPv6], etc. Commercial and open source DHCPv6 software often consider the extensions of DHCPv6 servers because they all cannot always meet the requirements that the users want. In this section, we introduce two typical DHCPv6 software: Cisco Prime Network Registrar and Kea DHCP, respectively.

### 3.2.1. Cisco Prime Network Registrar DHCP Server Extension APIs

Cisco Prime Network Registrar (CPNR) [CPNR] is an appliance which provides integrated Domain Name Server, DHCP, and IP Address Management services for IPv4 and IPv6. At the same time, CPNR DHCP server allows user to write extensions and functions to alter and customize how it handles and responds to DHCP requests. A network operator usually decides what packets process to modify, how to modify, and which extension point to attach the extension. Then the network operator just writes the extension and adds the well-written extension to the extension point of the DHCP server. Finally, the network operator reloads the DHCP server and can find that the server will do what he/her wants the server to do.

### 3.2.2. Kea DHCP Hook Mechanisms

Kea DHCP provides hook mechanisms, a defined interface for third-party or user-written code, to solve the problem that the DHCP server does not quite do what a network operator require. A network operator can use several well-defined framework functions to load and initialize a library and write specific callout functions to attach to the hook points. After building and configuring the hooks library, the server will do what the network operator require. Additionally, Kea DHCP allows users to use logging in the hooks library.

## 4. Problem Statement

This section elaborates the problem statement on multi-requirement extensions for DHCPv6. Section 4.1 describes the general model of DHCP, while Section 4.2 analyzes the unresolved problems and requirements, suggesting possible future work.

### 4.1. DHCP General Model

Figure 1 summarizes the DHCP general model and its possible extensions: DHCP messages, options, message processing functions, and address generation mechanisms.

```
 +-------------------+                            +-------------------+
 |   DHCPv6 client   |                            |   DHCPv6 relay    |
 | +---------------+ | DHCP messages with options | +---------------+ |
 | |   Message     | |<-------------------------->| |   Message     | |
 | |  processing   | |                            | |  processing   | |
 | |  functions    | |                            | |  functions    | |
 | +---------------+ |                            | +---------------+ |
 +-------------------+                            +-------------------+
                                                          ^
                                                          |
                                 DHCP messages with options |
                                                          |
                                                          V
                                                  +-------------------+
                                                  |   DHCPv6 server   |
     +------------+                               | +---------------+ |
     |  Address   |                               | |   Message     | |
     | generation |<------------------------------+-|  processing   | |
     | mechanisms |                               | |  functions    | |
     +------------+                               | +---------------+ |
                                                  +-------------------+
```

Figure 1: DHCP general model and its possible extensions.

### 4.2. Unresolved Problems

Currently, DHCPv6 protocol solves the problem that basic options are transmitted in plaintext. However, there are still many problems left to be resolved. Section 4.2.1, Section 4.2.2, Section 4.2.3, and Section 4.2.4 discuss these problems.

#### 4.2.1. DHCP Messages

People are always concerned about the security and privacy issues of DHCP protocol. [RFC7819] and [RFC7824] consider the privacy issues associated with the use of DHCPv4 and DHCPv6 by the Internet users, respectively. The current DHCP protocol does not resolve the problem that the options are transmitted in ciphertext. That is to say, any other nodes can see the options transmitted in the DHCPv6 messages between a DHCPv6 client and servers. Secure DHCPv6 [draft-ietf-dhc-sedhcpv6-21] considers using public cryptography to provide a deployable security mechanism, which can transmit basic options in DHCP messages exchanged between clients and servers. However, this draft is currently dead in IESG. In fact, new messages can be designed and added to DHCPv6 protocol, and DHCP messages can be exchanged in either plaintext or ciphertext.

#### 4.2.2. Options

In other cases, network operators may require DHCP messages to transmit some self-defined options between clients and servers. However, no such mechanisms in the current DHCP protocol support this requirement. DHCP clients do not allow users to transmit options not defined in standards, and not all DHCP server software support transmitting non-standardized options, either. For example, [NIDTGA] modifies the DHCPv6 message exchanges by adding some new options with cryptographic options. However, current DHCP standards do not provide related and flexible interfaces to meet such requirements. In fact, not only DHCP server software can provide interfaces for users to alter the way that they handle and respond to DHCP messages to meet their requirements, but DHCP client software can also provide such interfaces.

#### 4.2.3. Message Processing Functions

Although current commercial or open-source DHCP server software provide comprehensive functionality, they still cannot meet all customers' requirements of processing DHCP requests. Therefore, improved commercial or open-source DHCP server software will provide interfaces that customers can use to write their specific extensions to affect the way how DHCP servers handle and respond to DHCP requests. For example, not all networks prefer to use DHCPv6 servers to assign the privacy-preserving random-form addresses generated by some fixed address generation mechanism to DHCPv6 clients. Several address generation mechanisms for SLAAC [RFC4862] (e.g., IEEE 64-bit EUI-64 [RFC2464], Constant, semantically opaque [Microsoft], Temporary [RFC4941], and Stable, semantically opaque [RFC7217]) proposed for different requirements can be also utilized in DHCPv6 protocol. The many types of IPv6 address generation mechanisms available have brought about flexibility and diversity. Thus, network operators may alter their DHCPv6 servers through the given extensions to use their own preferred address generation mechanisms to assign addresses to DHCPv6 clients. However, not all DHCP software consider this extension.

#### 4.2.4. Address Generation Mechanisms

Different networks may prefer different address generation mechanisms. However, current DHCPv6 protocol only considers generating random IPv6 addresses. Corresponding interfaces should be open and defined to allow other address generation mechanisms to be configured.

#### 4.2.5. Extension Principles

The principles used to conduct multi-requirement extensions for DHCPv6 are summarized as follows:

1) Do not change the current DHCP general model.

2) Use simpler interfaces to define and support more extensions.

## 5. Security Considerations

Security issues related with DHCPv6 are described in Section 23 of [RFC3315].

Secure DHCPv6 [draft-ietf-dhc-sedhcpv6-21] attempts to provide a deployable security mechanism for end-to-end communication between DHCP clients and servers.

## 6. IANA Considerations

This document does not include an IANA request.

## 7. Acknowledgements

The authors would like to thank Bernie Volz, Tomek Mrugalski, Sheng Jiang, and Jinmei Tatuya for their comments and suggestions that

improved the [draft-ren-dhc-mredhcpv6]. Some ideas and thoughts of [draft-ren-dhc-mredhcpv6] are contained in this document.

## 8. References

### 8.1. Normative References

[CPNR] Cisco, "Cisco Prime Network Registrar", 2018, <https://www.cisco.com/c/en/us/products/cloud-systems-management/prime-network-registrar/index.html>.

[draft-ietf-dhc-sedhcpv6-21] Li, L., Jiang, S., Cui Y., Jinmei T., Lemon, T., and D. Zhang, "Secure DHCPv6", August 2017, <https://www.ietf.org/archive/id/draft-ietf-dhc-sedhcpv6-21.txt>.

[draft-ren-dhc-mredhcpv6] Ren, G., He, L., and Y. Liu, "Multi-requirement Extensions for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", March 2017.

[FreeRADIUS_DHCP] FreeRADIUS, "FreeRADIUS DHCP", 2017, <https://wiki.freeradius.org/features/DHCP>.

[GAGMS] Liu, Y., He, L., and G. Ren, "GAGMS: A Requirement-Driven General Address Generation and Management System", November 2017.

[ISC_DHCP] Internet System Consortium, "ISC DHCP", 2018, <http://www.isc.org/downloads/dhcp/>.

[Kea_DHCP] Internet System Consortium, "Kea DHCP", 2018, <https://www.isc.org/kea/>.

[kea_dhcp_hook_developers_guide] Internet Systems Consortium, "Hook Developer’s Guide", 2018, <https://jenkins.isc.org/job/Kea_doc/doxygen/df/d46/hooksdgDevelopersGuide.html>.

[Microsoft] Microsoft, "IPv6 interface identifiers", 2013, <https://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/sag_ip_v6_imp_addr7.mspx?mfr=true>.

[Microsoft_DHCP]
Microsoft, "Microsoft DHCP", 2008, <https://technet.microsoft.com/en-us/library/cc896553(v=ws.10).aspx>.

[NIDTGA] Liu, Y., Ren, G., Wu J., Zhang s., He, L., and Y. Jia, "Building an IPv6 address generation and traceback system with NIDTGA in Address Driven Network", 2015, <https://link.springer.com/article/10.1007/s11432-015-5461-0>.

[Nominum_DHCP]
Nominum, "Nominum DHCP", 2012, <https://www.nominum.com/press_item/nominum-releases-new-version-of-carrier-grade-dhcp-software-for-telecom-providers/>.

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

[RFC2464] Crawford, M., "Transmission of IPv6 Packets over Ethernet Networks", RFC 2464, DOI 10.17487/RFC2464, December 1998, <https://www.rfc-editor.org/info/rfc2464>.

[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, <https://www.rfc-editor.org/info/rfc3315>.

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

[RFC3898] Kalusivalingam, V., "Network Information Service (NIS) Configuration Options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 3898, DOI 10.17487/RFC3898, October 2004, <https://www.rfc-editor.org/info/rfc3898>.

[RFC4075] Kalusivalingam, V., "Simple Network Time Protocol (SNTP) Configuration Option for DHCPv6", RFC 4075, DOI 10.17487/RFC4075, May 2005, <https://www.rfc-editor.org/info/rfc4075>.

[RFC4242] Venaas, S., Chown, T., and B. Volz, "Information Refresh Time Option for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", RFC 4242, DOI 10.17487/RFC4242, November 2005, <https://www.rfc-editor.org/info/rfc4242>.

[RFC4649] Volz, B., "Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Relay Agent Remote-ID Option", RFC 4649, DOI 10.17487/RFC4649, August 2006, <https://www.rfc-editor.org/info/rfc4649>.

[RFC4704] Volz, B., "The Dynamic Host Configuration Protocol for IPv6 (DHCPv6) Client Fully Qualified Domain Name (FQDN) Option", RFC 4704, DOI 10.17487/RFC4704, October 2006, <https://www.rfc-editor.org/info/rfc4704>.

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

[RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6", RFC 4941, DOI 10.17487/RFC4941, September 2007, <https://www.rfc-editor.org/info/rfc4941>.

[RFC4994] Zeng, S., Volz, B., Kinnear, K., and J. Brzozowski, "DHCPv6 Relay Agent Echo Request Option", RFC 4994, DOI 10.17487/RFC4994, September 2007, <https://www.rfc-editor.org/info/rfc4994>.

[RFC5970] Huth, T., Freimann, J., Zimmer, V., and D. Thaler, "DHCPv6 Options for Network Boot", RFC 5970, DOI 10.17487/RFC5970, September 2010, <https://www.rfc-editor.org/info/rfc5970>.

[RFC6939] Halwasia, G., Bhandari, S., and W. Dec, "Client Link-Layer Address Option in DHCPv6", RFC 6939, DOI 10.17487/RFC6939, May 2013, <https://www.rfc-editor.org/info/rfc6939>.

[RFC7037] Yeh, L. and M. Boucadair, "RADIUS Option for the DHCPv6 Relay Agent", RFC 7037, DOI 10.17487/RFC7037, October 2013, <https://www.rfc-editor.org/info/rfc7037>.

[RFC7217] Gont, F., "A Method for Generating Semantically Opaque Interface Identifiers with IPv6 Stateless Address Autoconfiguration (SLAAC)", RFC 7217, DOI 10.17487/RFC7217, April 2014, <https://www.rfc-editor.org/info/rfc7217>.

[RFC7653] Raghuvanshi, D., Kinnear, K., and D. Kukrety, "DHCPv6 Active Leasequery", RFC 7653, DOI 10.17487/RFC7653, October 2015, <https://www.rfc-editor.org/info/rfc7653>.

[RFC7819] Jiang, S., Krishnan, S., and T. Mrugalski, "Privacy Considerations for DHCP", RFC 7819, DOI 10.17487/RFC7819, April 2016, <https://www.rfc-editor.org/info/rfc7819>.

[RFC7824] Krishnan, S., Mrugalski, T., and S. Jiang, "Privacy Considerations for DHCPv6", RFC 7824, DOI 10.17487/RFC7824, May 2016, <https://www.rfc-editor.org/info/rfc7824>.

[VitalQIP] Nokia, "Nokia VitalQIP", 2017, <https://networks.nokia.com/products/vitalqip-ip-address-management>.

[WIDE_DHCPv6] KAME project, "WIDE DHCPv6", 2008, <http://ipv6int.net/software/wide_dhcpv6.html>.

## 8.2. Informative References

[RFC3552] Rescorla, E. and B. Korver, "Guidelines for Writing RFC Text on Security Considerations", BCP 72, RFC 3552, DOI 10.17487/RFC3552, July 2003, <https://www.rfc-editor.org/info/rfc3552>.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 5226, DOI 10.17487/RFC5226, May 2008, <https://www.rfc-editor.org/info/rfc5226>.

[RFC7749] Reschke, J., "The "xml2rfc" Version 2 Vocabulary", RFC 7749, DOI 10.17487/RFC7749, February 2016, <https://www.rfc-editor.org/info/rfc7749>.

## Authors' Addresses

Gang Ren
Tsinghua University
Beijing  100084
P.R.China

Phone: +86-010 6260 3227
Email: rengang@cernet.edu.cn

Lin He
Tsinghua University
Beijing 100084
P.R.China

Email: he-l14@mails.tsinghua.edu.cn

Ying Liu
Tsinghua University
Beijing 100084
P.R.China

Email: liuying@cernet.edu.cn