

Dynamic Host Configuration (DHC)  
Internet-Draft  
Intended status: Informational  
Expires: November 24, 2019

G. Ren  
L. He  
Y. Liu  
Tsinghua University  
May 23, 2019

**Problem Statement of Multi-requirement Extensions for Dynamic Host  
Configuration Protocol for IPv6 (DHCPv6)  
draft-ietf-dhc-problem-statement-of-mredhcpv6-00**

Abstract

The manageability, security, privacy protection, and traceability of networks can be supported by extending DHCPv6 protocol according to requirements. This document analyzes current extension practices and typical DHCP server software on extensions, defines a DHCP general model, discusses some extension points, and present extension cases.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on November 24, 2019.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in [Section 4](#).e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

<a href="#">1.</a>	Introduction . . . . .	<a href="#">2</a>
<a href="#">2.</a>	Terminology . . . . .	<a href="#">3</a>
<a href="#">3.</a>	Current Extension Practices . . . . .	<a href="#">3</a>
3.1.	Standardized and Non-standardized DHCPv6 Extension Cases	3
<a href="#">3.2.</a>	Current DHCPv6 Server Software Cases . . . . .	<a href="#">4</a>
3.2.1.	Cisco Prime Network Registrar DHCP Server Extension APIs . . . . .	<a href="#">4</a>
<a href="#">3.2.2.</a>	Kea DHCP Hook Mechanisms . . . . .	<a href="#">4</a>
<a href="#">4.</a>	Problem Statement . . . . .	<a href="#">5</a>
<a href="#">4.1.</a>	DHCP General Model . . . . .	<a href="#">5</a>
<a href="#">4.2.</a>	Extension Discussion . . . . .	<a href="#">5</a>
<a href="#">4.2.1.</a>	DHCP Messages . . . . .	<a href="#">5</a>
<a href="#">4.2.2.</a>	Options . . . . .	<a href="#">6</a>
<a href="#">4.2.3.</a>	Message Processing Functions . . . . .	<a href="#">6</a>
<a href="#">4.2.4.</a>	Address Generation Mechanisms . . . . .	<a href="#">6</a>
<a href="#">4.2.5.</a>	Extension Principles . . . . .	<a href="#">7</a>
<a href="#">5.</a>	Extension Cases . . . . .	<a href="#">7</a>
<a href="#">6.</a>	Security Considerations . . . . .	<a href="#">8</a>
<a href="#">7.</a>	IANA Considerations . . . . .	<a href="#">8</a>
<a href="#">8.</a>	Acknowledgements . . . . .	<a href="#">8</a>
<a href="#">9.</a>	Normative References . . . . .	<a href="#">8</a>
	Authors' Addresses . . . . .	<a href="#">10</a>

## [1.](#) Introduction

The IP address plays a significant role in the communication of the Internet. IP address generation is also closely related to the manageability, security, privacy protection, and traceability of networks. Dynamic Host Configuration Protocol for IPv6 (DHCPv6) [[RFC8415](#)] is an important network protocol that can be used to dynamically provide IPv6 addresses and other network configuration parameters to IPv6 nodes. Actually, DHCPv6 continues to be extended and improved through new options, protocols or message processing mechanisms.

Although DHCPv6 provides more and more comprehensive functionalities and DHCPv6 server software also provides extension interfaces to allow administrators to alter and customize the way how they handle and respond to DHCPv6 messages, there is still a lack of a general insight into where and how to conduct extensions in DHCPv6 effectively. The extensions to DHCPv6 can be various according to multiple requirements. Therefore, a detailed analysis is required to clarify the problems, design principles, and extract and unify the



design specifications to help better solve the multi-requirement extension problems.

In summary, multi-requirement extensions for DHCPv6 can be conducted to support the administrator's self-defined functionalities. As DHCPv6 is an important and useful protocol related to IPv6 addresses generation, it can provide more extended and flexible functionalities to meet administrators' 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 administrators modify the open-source DHCP servers to solve their problems. However, a great amount of time will be taken to understand the open source DHCP server codes, 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 codes of the server software may change (new functionalities 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 administrators' specific problems are very necessary and significant.

This document describes current extension practices and typical DHCP server software on extensions and provides a problem statement by defining a DHCP general model, discussing the extension problems, and presenting extension cases.

## **2. Terminology**

Familiarity with DHCPv6 and its terminology, as defined in [[RFC8415](#)], is assumed.

## **3. Current Extension Practices**

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

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

Extended options	Most extensions for DHCPv6 are implemented in this way. New-defined options carry specific parameters in DHCPv6 messages, which helps DHCPv6 clients or servers know the detailed situation with each other.
------------------	--



- Extended messages    Some documents define new protocols that aim to achieve specific goals, e.g., active leasequery [[RFC7653](#)], GAGMS [[GAGMS](#)].
- Extended entities    Some documents introduce third-party entities into the communications of DHCPv6 to achieve specific goals and provide better services, e.g., authentication [[RFC7037](#)].

### **3.2. Current DHCPv6 Server Software Cases**

A lot of commercial and open source DHCP servers 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](#)], and DHCP Broadband [[DHCP Broadband](#)]. Commercial and open source DHCPv6 software often considers the extensions of DHCPv6 servers because they cannot always meet the requirements that the administrators want. In this section, we introduce two typical DHCPv6 servers: Cisco Prime Network Registrar and Kea DHCP.

#### **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 provides extension APIs and allows administrators to write extensions and functions to alter and customize how it handles and responds to DHCP requests. A network operator usually decides what packet 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 debugs whether the server runs as it expects.

#### **3.2.2. Kea DHCP Hook Mechanisms**

Kea DHCP provides hook mechanisms, a well-designed interface for third-party 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 runs as the network operator requires. Additionally, Kea DHCP allows the network operator to use logging in the hooks library.



## 4. Problem Statement

This section elaborates the problem statement of multi-requirement extensions for DHCPv6. [Section 4.1](#) describes the general model of DHCP, while [Section 4.2](#) analyzes the extension points 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.

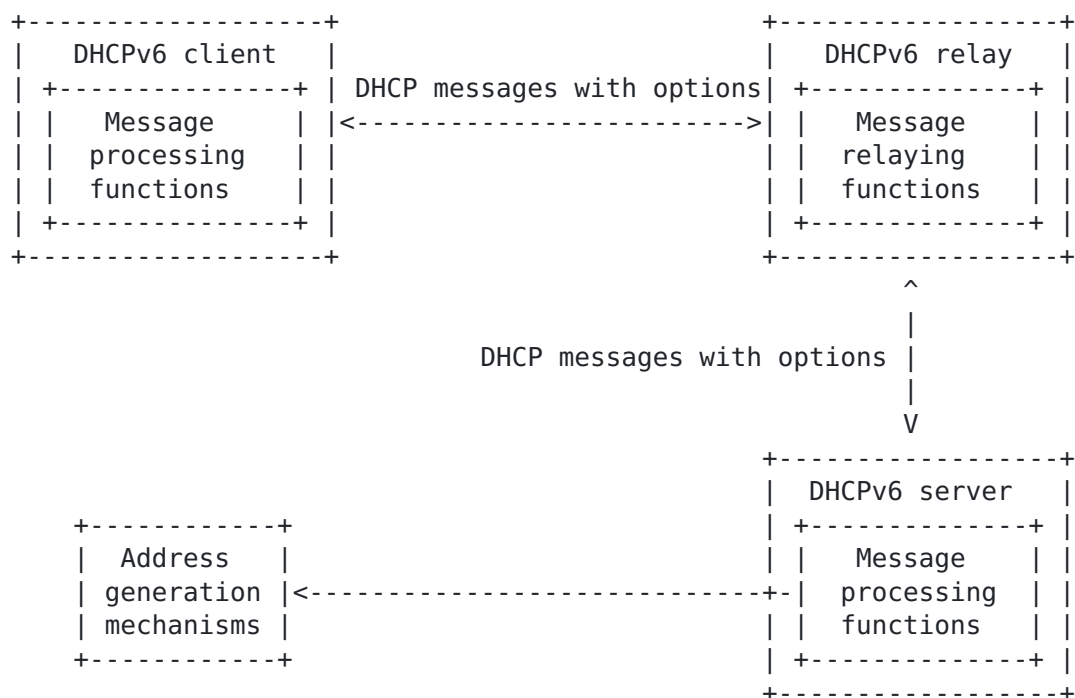


Figure 1: DHCP general model and its possible extensions.

## 4.2. Extension Discussion

### 4.2.1. DHCP Messages

In fact, new messages can be designed and added to DHCPv6 protocol, e.g., active leasequery [[RFC7653](#)]. But currently, people are concerned about the security and privacy issues of DHCP protocol. [[RFC7819](#)] and [[RFC7824](#)] describe the privacy issues associated with the use of DHCPv4 and DHCPv6, respectively. DHCPv6 does not provide the privacy protection on messages and options. That is to say,





other nodes can see the options transmitted in the DHCPv6 messages between DHCPv6 clients and servers.

#### **4.2.2. Options**

DHCPv6 allows defining options for common requirements, e.g., DNS and NTP. In other cases, network operators may require DHCP messages to transmit some self-defined options between clients and servers. Currently, vendor-specific information option allows clients and servers to exchange vendor-specific information. Therefore, administrative domains can define and use sub-options of vendor-specific option to serve their private purposes. The content of the self-defined options may come from two sources: devices and users. If the content of self-defined options comes from users, two methods can be used to solve the problem. The first one is that the clients provide related interfaces to receive such information, which is currently merely supported. The second one is that DHCPv6 relays obtain such information and add it into the clients' requests. But this always depends on other protocols to allow DHCPv6 relays to get the information first.

#### **4.2.3. Message Processing Functions**

Although current commercial or open-source DHCP server software provides comprehensive functionalities, they still cannot meet all customers' requirements of processing DHCP requests. Therefore, they 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 utilized in DHCPv6 protocol as well. 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 considers this extension.

#### **4.2.4. Address Generation Mechanisms**

Currently, the DHCPv6 servers assign addresses, prefixes and other configuration options according to their configured policies. Generally, different networks may prefer different address generation



mechanisms. Corresponding interfaces could 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. Extension Cases**

Administrative domains may enforce local policies according to their requirements, e.g., authentication, accountability. Several kinds of multi-requirement extensions are presented in this section, including configurations in current DHCP software, option definition and server modification, and message definition between DHCP entities and third-party entities.

Currently, many DHCP servers provide administrative mechanisms, e.g., host reservation and client classification. Host reservation is often used to assign certain parameters (e.g., IP addresses) to specific devices. Client classification is often used to differentiate between different types of clients and treat them accordingly in certain cases.

To meet specific requirements, more complicated extensions of DHCPv6 are needed. For example, considering such a requirement that DHCP servers assign IP addresses generated by user identifiers to the clients in a network to hold users accountable, two extensions should be fulfilled to meet this requirement. The first one is that clients send their user identifiers to servers. This can be achieved by defining and using sub-options of vendor specific information option. The second one is that servers use user identifiers to generate IP addresses. To achieve this goal, extension mechanisms provided by the server software such as extension points in CPNR [[CPNR](#)] and hook mechanisms in Kea DHCP [[Kea DHCP](#)] can be used.

Some extensions for DHCPv6 may need the support of third-party entities. For example, [[RFC7037](#)] introduces RADIUS entities into the message exchanges between DHCPv6 entities for better service provision. In fact, the authentication in [[RFC7037](#)] can be also used to meet the accountability requirement mentioned above because it is important to authenticate users first before assigning IP addresses generated from user identifiers. Usually, this kind of extension



requires the definition of messages communicated between DHCP entities and third-party entities, e.g., active leasequery [[RFC7653](#)].

IPv6 addresses are related to manageability, security, traceability, and accountability of networks. As DHCPv6 assigns IPv6 addresses to IPv6 nodes, it is important that DHCPv6 provides interfaces to allow administrative domains to conduct extensions to meet their multi-requirements.

## 6. Security Considerations

Security issues related with DHCPv6 are described in [Section 22 of \[RFC8415\]](#).

## 7. IANA Considerations

This document does not include an IANA request.

## 8. 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.

## 9. 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>>.
- [DHCP\_Broadband] Weird Solutions, "DHCP Broadband", 2018, <<https://www.weird-solutions.com/carrier-solutions/dhcp-broadband>>.
- [[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](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.msp?mfr=true](https://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/sag_ip_v6_imp_addr7.msp?mfr=true)>.

## [Microsoft\_DHCP]

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

## [Nominum\_DHCP]

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

[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>>.

[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>>.

[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>>.
- [RFC8415] Mrugalski, T., Siodelski, M., Volz, B., Yourtchenko, A., Richardson, M., Jiang, S., Lemon, T., and T. Winters, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)", [RFC 8415](#), DOI 10.17487/RFC8415, November 2018, <<https://www.rfc-editor.org/info/rfc8415>>.
- [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](http://ipv6int.net/software/wide_dhcpv6.html)>.

#### Authors' Addresses

Gang Ren  
Tsinghua University  
Beijing 100084  
P.R.China

Phone: +86-010 6260 3227  
Email: [rengang@cernet.edu.cn](mailto: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