CHAP-based Authentication for DHCPv6
draft-ota-dhc-auth-chap-00.txt

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

This Internet-Draft will expire on December 30, 2005.

Copyright Notice

Copyright (C) The Internet Society (2005). All Rights Reserved.

Abstract

In delayed authentication in DHCPv6[RFC3315], the hardware identifier is used to select the key, and the key is exchanged between the server and the client in advance. Therefore, when a user tries to use a new device, it is necessary to add key information to the new device and DHCPv6 servers.

In this document, we propose a procedure for introducing CHAP[RFC1994] into the DHCPv6 authentication procedure. This procedure allows the client get the host configuration information.
related to the user and exchanges a secret key to use in later sequence.

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.

Most of the terms used in this draft are defined in RFC3315.

Table of Contents

1. Introduction..........................3
2. Issues...................................3
3. Protocol Overview........................4
   4.1. Authentication Information in Advertise Messages........7
   4.2. Authentication Information in Request Messages........7
   4.3. Authentication Information in Reply Messages corresponding to Request Message..........................8
5. Client and Server Considerations..................9
   5.1. Client Considerations..................9
      5.1.1. Sending Solicit Messages..................9
      5.1.2. Receiving Advertise Messages................9
      5.1.3. Sending Request Messages..................9
      5.1.4. Sending Confirm, Renew, Rebind, Decline or Release Messages..........................9
      5.1.5. Sending Information-Request Messages.............9
      5.1.6. Receiving Reply Messages..................10
      5.1.7. Receiving Reconfigure Messages................10
      5.1.8. When the key life time is over................10
   5.2. Server Considerations..................10
      5.2.1. Receiving Solicit Messages and Sending Advertise Messages..........................10
      5.2.2. Receiving Request Messages and Sending Reply Messages..........................10
      5.2.3. Receiving Confirm, Renew, Rebind, Decline or Release Messages..........................11
      5.2.4. Sending Reply Messages in Information-Request/Reply sequence..........................11
6. Security Considerations........................11
7. IANA Consideration........................11
8. Informative Reference........................11
Author’s Addresses........................11
Intellectual Property Statement..................12
1. Introduction

In delayed authentication, which is the authentication procedure described in RFC3315, the hardware identifier, such as the MAC address, is used to select the key, and it is assumed that the key is exchanged between the server and the client in advance. Therefore, when a user tries to use a new device, it is necessary to submit new hardware identifiers and new key information to the DHCPv6 server. We think that it will be deployment issue.

To avoid this problem, we propose a user based authentication procedure. In this procedure, DHCPv6 server authenticates user using CHAP mechanism during solicitation procedure. To identify users, we assumed that user identifier, such as FQDN of user, is used as identifier of peer. And shared secret, which is used to establish security association between DHCPv6 client and server for subsequent sequences, is exchanged using Diffie-Hellman mechanism.

2. Issues

In the Delayed Authentication procedure, the DHCPv6 server selects a key based on the hardware identifier of the DHCPv6 client, such as MAC address, and it is assumed that keys are exchanged between the server and the client in advance. Therefore, when a user tries to use a new device, it is necessary to add the key to the new device and DHCPv6 servers. We think that there will be the following deployment issues:

-The user is required to submit a new hardware identifier and a new key to the operator of the DHCPv6 server.

The original purpose of DHCPv6 authentication was to avoid server and client spoofing in general environments such as a LAN, so the DHCPv6 authentication procedure is required to provide mutual authentication. On the other hand, we can assume that nobody can spoof the DHCPv6 server in a commercial network. In such a network, we think that it is unnecessary for the client to authenticate the server. A network access authentication procedure, such as CHAP, is one of the major one-way authentication procedures. A lot of network access procedures use the user identifier to select a secret. Therefore, an AAA server is required to manage one secret per user even if users change terminal.
Therefore, we propose a new authentication procedure that applies an access authentication method based on the user identifier to the DHCPv6 authentication procedure.

3. Protocol Overview

Here, we introduce a new authentication procedure that uses CHAP as the authentication method. This procedure allows the server to execute authentication based on the user identifier. We call it the CHAP-based Authentication Procedure. An example sequence of this procedure is illustrated in Figure 1.

When the DHCPv6 server receives a Solicit message, the server replies with the Advertise message, which carries the challenge and identifier. The client calculates the Response value using the challenge and a pre-shared secret such as a password. And the client sends a Request message with the user-identifier, identifier, and Response value. When the DHCPv6 server receives a Request message, it selects a challenge based on the identifier. The DHCPv6 server may send an authentication request message to the AAA server, and the AAA server selects a secret based on the user identifier and evaluates the Response value. But the protocol between the DHCPv6 server and the AAA server is out of our scope. If the evaluation has been completed successfully, the DHCPv6 server sends a Reply message to the DHCPv6 client.
4. Packet Format of the Authentication Option in the CHAP-based Authentication Procedure

In a Solicit message, the client fills in the protocol, algorithm, and RDM fields in the Authentication option with the client's preferences. The client sets the replay detection field to zero and omits the authentication information field. The client sets the option-len field to 11.

In all other messages, the protocol and algorithm fields identify the procedure used to construct the contents of the authentication information field. The RDM field identifies the procedure used to construct the contents of the replay detection field.

The format of the Authentication information for CHAP-based authentication procedure is:

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    MSG_TYPE   |   Identifier  |         MSG_Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Attribute                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         .........                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

**MSG TYPE**

The MSG TYPE identifies the type of message. We set the following types.

1 Challenge
2 Response
3 Success
4 Failure

**Identifier**

The Identifier shows that it is a series of a CHAP sequence. When a Challenge value is generated, the identifier is generated at random by DHCPv6 Server.
MSG_Length
The total size of the authentication information field. The value contains MSG TYPE, Identifier, MSG length, and Attribute field length.

Attribute
The Attribute carries authentication information. Details of the format in the Attribute field are shown below.

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
+-----------------------------------------------+
| TYPE | Length | Value ...
+-----------------------------------------------+

TYPE
The TYPE of Attribute shows the type of information of Value.
1 User-Name
3 CHAP-Password
60 CHAP-Challenge
80 DH-group
81 KeyExchange
82 DHCP realm
83 key ID
84 life time of the key

Length
Length of Value in octets.

Value
Value of Attribute related to TYPE number.
Information corresponding to the TYPE number buried under the Value field is shown below.

<table>
<thead>
<tr>
<th>TYPE</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>String of User-Name and domain tied by using @.</td>
</tr>
<tr>
<td>3</td>
<td>Response value</td>
</tr>
<tr>
<td>60</td>
<td>Challenge value</td>
</tr>
<tr>
<td>80</td>
<td>Diffie-Hellman group value</td>
</tr>
<tr>
<td>81</td>
<td>Diffie-Hellman public key value</td>
</tr>
<tr>
<td>82</td>
<td>DHCP realm info</td>
</tr>
<tr>
<td>83</td>
<td>key Identifier</td>
</tr>
<tr>
<td>84</td>
<td>Exchanged key's life time</td>
</tr>
</tbody>
</table>
4.1. Authentication Information in Advertise Messages

The Advertise message has the following authentication information field.

```
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Challenge(1)  |  Identifier   |         MSG_Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Type(60)     |     Length    |         Challenge value       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

MSG TYPE field is 1 (Challenge). The values of Identifier and Challenge value are generated by the server. This authentication information has an attribute, Type(60). The attribute Type 60 is filled with the Challenge value.

4.2. Authentication Information in Request Messages

A Request message has the following authentication information field.

```
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Response(2)  |  Identifier   |         MSG_Length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type(1)   |     Length    |       User-Name@domain        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type(3)   |     Length    |         Response value        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   DH-group(80)  |     Length    |       DH-group value          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| KeyExchange(81) |     Length    |        public key value       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

MSG TYPE field is 2 (Response). Identifier is the same as the one included in the Advertise message. This authentication information has two attributes: Type (1) and Type (3). The attribute Type 1 is filled with User-Name@domain, which the client has beforehand. The attribute Type 3 is filled with Response value, which is calculated by the client. The attribute Type 80 is filled with DH-group value. The attribute Type 81 is filled with public key value generated by the client, which uses to calculate secret key with the server.
4.3. Authentication Information in Reply Messages corresponding to Request Message

The server selects authentication information included in the Reply message according to the authentication information received from the AAA server. When the server receives the message of authentication success, it sends a Reply message containing authentication information with the MSG_TYPE field set to 3 (Success). The attribute Type 80 is filled with DH-group value. The attribute Type 81 is filled with public key value generated by the server, which uses to calculate secret key by the client. Additionally, the message includes other necessary information (DHCP realm, key ID, life time of the key).

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Success    |  Identifier   |     MSG_Length (4[Octets])    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| DH-group(80)  |     Length    |       DH-group value          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|KeyExchange(81)|     Length    |        public key value       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| DHCP realm(82)|     Length    |        DHCP realm info        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   key ID(83)  |     Length    |         key identifier        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| life time(84)|     Length    |   Exchanged key's life time   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

When the server receives the message of the authentication failure, the server sends the Reply message with the MSG TYPE field set to 4 (Failure).

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Failure    |  Identifier   |     MSG_Length (4[Octets])    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The server that receives an authentication failure can be allowed to send a Reply message that does not include other options.
5. Client and Server Considerations

5.1. Client Considerations

The client announces its intention to use DHCPv6 authentication by including an authentication option in its Solicit message.

5.1.1. Sending Solicit Messages

When the client demands to authenticate, a Solicit message includes an authentication option with the desired protocol, algorithm, and RDM. The client does not include any replay detection or authentication information in the authentication option.

5.1.2. Receiving Advertise Messages

The client validates Advertise messages. If the Advertise messages have a Challenge value in the authentication information field, the client generates a Response value by applying MD5 to the data connecting Identifier, the secret that the client has beforehand, and the Challenge value.

5.1.3. Sending Request Messages

The client sends a Request message with User-Name as the user identifier, Identifier, Response value, DH-group, and KeyExchange attribute in the authentication information fields. A public key value in KeyExchange attribute is calculated by the client using DH-group which received in Advertise message.

5.1.4. Sending Confirm, Renew, Rebind, Decline or Release Messages

If the client received Reply messages before, it can send Confirm, Renew, Decline or Release Messages with authentication information using generated key. In this time, the client follows the sequence at section 21.4.4.3 in RFC3315. It is different only to put the number of this authentication in the protocol field in the Authentication option.

5.1.5. Sending Information-Request Messages

In this document, we support only stateful DHCP. Because, this procedure assumes use to the situation which is disbursed the address to the client, and managed by the server. The client follows the sequence at section 21.4.4.3 in RFC3315 using the secret key which the client gets in section 5.1.6. It is different only to put the
number of this authentication in the protocol field in the Authentication option.

5.1.6. Receiving Reply Messages

If the messages which client received include "Success" in authentication information, the client draws the public key value in KeyExchange Attribute. And the client calculates secret key by Diffie-Hellman algorism.

5.1.1. Receiving Reconfigure Messages

The client follows the sequence at section 21.4.4.6 in RFC3315. It is different only to put the number of this authentication in the protocol field in the Authentication option.

5.1.2. When the key life time is over

The client sends Solicit Message and start solicitation procedure to exchange new secret key. It is different only to put the number of this authentication in the protocol field in the Authentication option.

5.2. Server Considerations

5.2.1. Receiving Solicit Messages and Sending Advertise Messages

A server that receives a Solicit message with the protocol field set to CHAP-based authentication procedure value generates a Challenge value and Identifier. The server sends them with Advertise Messages. The server MUST record the identifier related to the Challenge value during the authentication procedure.

5.2.2. Receiving Request Messages and Sending Reply Messages

If the Request message includes authentication option which type is set with CHAP Authentication, the receiving server chooses a Challenge value related to the Identifier and sends the User-Name, Identifier, Challenge value, and Response value to the AAA server. After the server has received the authentication result from the AAA server, it sends the client a Reply message containing the authentication option, which includes success or failure in MSG TYPE and KeyExchange attribute which include public key value generated by the server and other information. And the server get secret key of the client calculated by received public key value from the client.
5.2.3. Receiving Confirm, Renew, Rebind, Decline or Release Messages

The client follows the sequence at section 21.4.5.2 in RFC3315. It is different only to put the number of this authentication in the protocol field in the Authentication option.

5.2.4. Sending Reply Messages in Information-Request/Reply sequence

The client follows the sequence at section 21.4.5.2 in RFC3315. It is different only to put the number of this authentication in the protocol field in the Authentication option.

6. Security Considerations

The CHAP-based authentication procedure gives the procedure for authenticating the client. This protocol does not give the means of server authentication, so it has a weakness in terms of clarifying the server.

7. IANA Consideration

This document introduces a new authentication mechanism. The type numbers for the protocol field in the authentication option are currently TBD. An appropriate request will be made to IANA if this Internet draft gets accepted as an RFC.

8. Informative Reference


Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2005).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.
Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.