Network Working Group T. Jinmei
Internet-Draft Toshiba
Expires: January 10, 2005 July 12, 2004

Clarifications on DHCPv6 Authentication
draft-jinmei-dhc-dhcpv6-clarify-auth-00.txt

Status of this Memo

Copyright Notice

Abstract

This document describes issues about the DHCPv6 authentication mechanism identified from implementation experiences. It also tries to propose resolutions to some of the issues.

1. Introduction

Several questions arose on the authentication mechanism of DHCPv6 [RFC3315] from implementation experiences, particularly on its delayed authentication protocol. Some of the questions may require a change or addition to the current protocol, and one of them may even

cause discussions on a security threat.

This document describes the issues based on the questions, and tries to propose resolutions for some of them, hoping the resolutions will be merged, if valid and accepted, to the next version of the base specification.

2. Usage with Information-Request

According to [RFC3315], it seems possible to use the authentication mechanism for Information-request and Reply exchanges. The RFC says in Section 21.4.4.4 as follows:

> If the server has selected a key for the client in a previous message exchange (see section 21.4.5.1), the client MUST use the same key to generate the authentication information throughout the session.

However, this description is not really clear. Section 21.4.5.1, which is referred from the above part, actually describes the case of Solicit and Advertise exchange:

> 21.4.5.1. Receiving Solicit Messages and Sending Advertise Messages
>
> The server selects a key for the client and includes authentication information in the Advertise message returned to the client as specified in section 21.4. [...]

It does not necessarily mean contradiction because the client and the server may have exchanged Solicit and Advertised with authentication before starting the Information-request and Reply exchange. However, it then restricts the usage scenario of the authentication mechanism for Information-request and Reply exchanges. In particular, this assumption prohibits the use of the mechanism with the "stateless" service using DHCPv6 [RFC3736]. Whereas the specification allows an implementation that only supports the stateless service and does not support Solicit and Advertise messages, the authentication mechanism depends on Solicit and Advertise exchanges.

This fact can (partly) invalidate a security consideration in [RFC3736]:

> Authenticated DHCP, as described in sections 21 and 22.11 of the DHCP specification [1], can be used to avoid attacks mounted through the stateless DHCP service.

(where [1] refers to [RFC3315].) In fact, as was just shown above,

authenticated DHCP cannot be used unless the implementations also support Solicit and Advertise messages (or the entire [RFC3315] in general).

It should also be noted that [RFC3315] does not define how the server should do when it receives an Information-request message containing an authentication option; Section 21.4.5.2 excludes the Information-request message.

2.1 Suggested Resolution

Considering the fact that [RFC3736] allows implementations that only support the subset of the full specification [RFC3315], it should make sense to define the authentication usage for Information-request and Reply exchanges separately.

One significant difference between Information-request and other "stateful" cases is that there is no explicit notion of "session" in the former. In some cases, however, the same client and server may exchange Information-request and Reply multiple times, where the entire exchanges can be regarded as a "session". For example, the client may want to get different configuration information in multiple exchanges. Also, if the client and the server use the lifetime option, [I-D.ietf-dhc-lifetime] they will restart exchanges when the lifetime expires.

The proposed revision of Section 21.4.4.4 is therefore as follows:

> 21.4.4.4. Sending Information-request Messages
>
> When the client sends an Information-request message and wishes to use authentication, it includes an Authentication option with the desired protocol, algorithm and RDM as described in section 21.4. The client does not include any replay detection or authentication information in the Authentication option.
>
> If the client authenticated past exchanges of Information-request and Reply, the client MAY reuse the same key used in the previous exchanges to generate the authentication information. In this case, the client generates authentication information for the Information-request message as described in section 21.4.
>
> Note that the keys used for these exchanges are separately managed from the keys used for the other exchanges beginning with the Solicit message when the two types of exchanges run concurrently, while the two keys may happen to be the same. For example, replay detection should be performed separately, and validation failure for one type of exchanges does not affect the other.

Section 21.4.4.5 will also need to be revised. However, since this section has a separate issue per se as will be discussed in Section 6, we do not discuss further details on this here.

The server side behavior needs to be described, too. Along with the change to Section 21.4.4.4 above, we propose to add a new subsection of Section 21.4.5:

21.4.5.x. Receiving Information-request Messages and Sending Reply Messages

If the Information-request message includes an authentication option without authentication information, the server selects a key for the client and includes authentication information in the Reply message returned to the client as specified in section 21.4. The server MUST record the identifier of the key selected for the client so that it can validate further Information-request messages from the client if the client reuses the same key for the future exchanges.

If the Information-request message includes an authentication option with authentication information, the server uses the key identified in the message and validates the message as specified in section 21.4.2. If the message fails to pass the validation test, the key identified by the authentication information of the message is not identical to the key that the server used in the previous exchange, or the server has not recorded a key for the client, the server MUST discard the message and MAY choose to log the validation failure.

If the message passes the validation test, the server responds to the Reply message as described in section 18.2.5. The server MUST include authentication information generated using the key just selected or identified in the received message, as specified in section 21.4.

Note that the keys used for these exchanges are separately managed from the keys used for the other exchanges beginning with the Solicit message when the two types of exchanges run concurrently (See Section 21.4.4.4).

3. What If Replay Is Detected

It is not clear what the receiver should do when an attempt of replay attack is detected from either Section 21.3 or Section 21.4.2 of [RFC3315].

3.1 Suggested Resolution

It should be natural to discard a DHCP message containing an authentication option whose replay detection field indicates a replay attack.

Instead of concentrating on this particular case, we propose to revise the entire second paragraph of Section 21.4.2 as follows:

> To validate an incoming message, the receiver first checks that the value in the replay detection field is acceptable according to the replay detection method specified by the RDM field. If no replay is detected, then the receiver computes the MAC as described in [8]. The entire DHCP message (setting the MAC field of the authentication option to 0) is used as input to the HMAC-MD5 computation function. If the MAC computed by the receiver matches the MAC contained in the authentication option, the message regarded as valid. If the above procedure fails at any stage, the receiver MUST discard the DHCP message.

4. Definition of Unauthenticated Messages

[RFC3315] uses the phrase of "unauthenticated message(s)" in Sections 21.4.4.2 and 21.4.4.5 without formally defining the term. A reasonable interpretation of the phrase is probably as follows: a DHCPv6 message is called unauthenticated when it fails the validation test described in Section 21.4.2, it does not contain an authentication option, or when it includes an authentication option but does not have authentication information when necessary.

In this document, we assume the above interpretation.

5. Inconsistent Behavior for Unauthenticated Messages

[RFC3315] says in Section 21.4.2 (Message Validation) as follows:

> If the MAC computed by the receiver does not match the MAC contained in the authentication option, the receiver MUST discard the DHCP message.

On the other hand, Section 21.4.4.2 allows the client to respond to an Advertise even if it fails to authenticate the message:

> Client behavior, if no Advertise messages include authentication information or pass the validation test, is controlled by local policy on the client. According to client policy, the client MAY choose to respond to an Advertise message that has not been authenticated.

This seems to say, for example, that the client MAY accept an Advertise message based on its local policy, even if the MAC computed by the receiver does not match the MAC contained in the authentication option. Apparently this contradicts with the requirement in Section 21.4.2.

5.1 Suggested Resolution

There seems to be no valid reason for accepting an Advertise message if it fails validation. On the other hand, it may make sense in some cases that the client accepts the other type of unauthenticated messages, that is, messages that do not include an authentication option.

The suggested change to Section 21.4.4.2 is thus as follows. We use a new term "non-authenticated messages" meaning DHCPv6 messages that do not contain an authentication option.

> [...]
>
> Client behavior, if no Advertise messages include authentication information is controlled by local policy on the client. According to client policy, the client MAY choose to respond to a non-authenticated Advertise message.
>
> The decision to set local policy to accept non-authenticated messages should be made with care. Accepting a non-authenticated Advertise message can make the client vulnerable to spoofing and other attacks. If local users are not explicitly informed that the client has accepted a non-authenticated Advertise message, the users may incorrectly assume that the client has received an authenticated address and is not subject to DHCP attacks through non-authenticated messages.
>
> A client MUST be configurable to discard non-authenticated messages, and SHOULD be configured by default to discard non-authenticated messages if the client has been configured with an authentication key or other authentication information. If a client does accept a non-authenticated message, the client SHOULD inform any local users and SHOULD log the event.

The second paragraph of Section 21.4.4.5 also needs a change:

> If the client accepted a non-authenticated Advertise message, the client MAY accept a non-authenticated Reply message from the server.

If we take this suggestion, then we will not need the notion of

"unauthenticated message". As a result, the issue described in Section 4 will become a non issue.

6. Possibility of Dos Attack

Section 21.4.4.5 of the RFC says as follows:

> If the Reply fails to pass the validation test, the client MUST restart the DHCP configuration process by sending a Solicit message.

The purpose of this specification is probably to avoid a deadlock scenario when the server suddenly reboots forgetting the authentication key and/or the replay detection counter.

However, this behavior can easily cause denial of service (DoS) attacks; the attacker can simply send an invalid Reply message at some valid timing and can invalidate configuration information of the client or can prevent the client from configuring itself.

As a side issue, this section seems to not consider Information-request and Reply exchanges.

6.1 Discussion on Resolution

Even if a Reply message does not pass the validation tests, it is probably reasonable to wait for an authenticated one until the first timeout. Additionally, if the Reply message is a response to Release, the client will not have to restart the configuration process by Solicit. It can simply quit the session when the first timeout occurs.

Reply messages to Information-request will need a separate consideration. Obviously, it does not make sense to send a Solicit message when the validation tests for a Reply message to Information-request fail. The appropriate behavior is probably to resend an Information-request message without including authentication information based on the key previously used, and to restart authentication.

7. Lack of Authentication from Client

It is not clear what the server should do when the client does not include an authentication option while the server has previously sent authentication information in the same session.

For messages other than Information-request, the appropriate behavior depends on the resolution for the issue discussed in Section 5.

Assuming the proposed resolution is adopted, the server should discard the message, since the client should have accepted the key as long as it is valid and then must use the key for succeeding message according to Section 21.4.4.3 of [RFC3315].

The appropriate behavior for Information-request depends on the resolution discussed in Section 2. If we take the proposed resolution, then the server should accept the message and select a new key, which may be the same as the one used before though, for the new exchanges as described in Section 2.

8. Key Consistency

[RFC3315] requests in Section 21.4.4.3 that the client use the same key used by the server to generate the authentication information. However, it is not clear from the RFC what the server should do if the client breaks this rule. It says in Section 21.4.5.2 that

> If the message [...] or the server does not know the key identified by the 'key ID' field, the server MUST discard the message and MAY choose to log the validation failure.

It is not clear whether "does not know the key" means a different key from the one the server specified in the Advertise message. If this is the intent, this sentence should be clarified as follows:

> If the message [...] or the key identified by the authentication information of the message is not identical to the key that the server has been using in the session, the server MUST discard the message and MAY choose to log the validation failure.

9. Security Considerations

This document specifically talks about security issues for DHCPv6. It also points out a possibility of DoS attacks, and gives some considerations on how to prevent them.

10. IANA Considerations

This document has no actions for IANA.

11. References

11.1 Normative References

[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.

[RFC3736] Droms, R., "Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6", RFC 3736, April 2004.

11.2 Informative References

[I-D.ietf-dhc-lifetime]
Venaas, S. and T. Chown, "Lifetime Option for DHCPv6", draft-ietf-dhc-lifetime-00 (work in progress), March 2004.

Author's Address

Tatuya Jinmei
Corporate Research & Development Center, Toshiba Corporation
1 Komukai Toshiba-cho, Saiwai-ku
Kawasaki-shi, Kanagawa 212-8582
Japan

Phone: +81 44-549-2230
EMail: jinmei@isl.rdc.toshiba.co.jp

Intellectual Property Statement

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

Acknowledgment

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