Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: May 17, 2017

T. Mizrahi
Marvell
D. Mayer
H. Stenn
Network Time Foundation

November 13, 2016

Network Time Protocol Version 4 (NTPv4) Extension Fields
draft-stenn-ntp-extension-fields-00

Abstract

Network Time Protocol version 4 (NTPv4) defines the optional usage of extension fields. An extension field, as defined in RFC 5905 [RFC5905], resides after the end of the NTP header, and supplies optional capabilities or information that is not conveyed in the standard NTP header. This document updates RFC 5905 [RFC5905] by clarifying some points regarding NTP extension fields and their usage with legacy Message Authentication Codes (MACs).

With the adoption of this update, the authors recommend rescinding [Err3627].

Table of Contents

1. Introduction

The NTP header format consists of a set of fixed fields that may be followed by optional fields. Two types of optional fields are defined: extension fields as defined in Section 7.5 of RFC 5905 [RFC5905], and legacy Message Authentication Codes (legacy MACs).

If a legacy MAC is used, it resides at the end of the packet. This field can be either a 4-octet crypto-NAK or data that is usually 20 or 24 octets long.

NTP extension fields are defined in RFC 5905 [RFC5905] as a generic mechanism that allows the addition of future extensions and features without modifying the NTP header format (Section 16 of RFC 5905 [RFC5905]).

Section 7.5 of RFC 5905 [RFC5905] clearly states that "one or more extension fields can be inserted after the header and before the MAC, which is always present when an extension field is present." However, the experimental Checksum Complement RFC 7821 [RFC7821] cannot be used if the NTP packet contains a MAC.

To allow for extension fields that do not require a MAC, changes to the NTPv4 specification must be made. [Err3627] was an attempt to

clarify the rules around MACs, but with the adoption of this proposal the authors recommend rescinding [Err3627].

This document better specifies and clarifies both Extention Fields and the requirements and parsing of a legacy MAC, with changes to address errors found after the publication of RFC 5905 [RFC5905] with respect to extension fields. Specifically, this document updates Section 7.5 of RFC 5905 [RFC5905], clarifying the relationship between extension fields and MACs, and defines the behavior of a host that receives an unknown extension field.

2. Conventions Used in This Document

2.1. Requirements Language

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 [RFC2119].

2.2. Terms and Abbreviations

MAC - Message Authentication Code

NTPv4 - Network Time Protocol, Version 4 RFC 5905 [RFC5905]

3. NTP Extension Fields - RFC 5905 Update

This document updates Section 7.5 of RFC 5905 [RFC5905] as follows:

3.1. OLD: 7.5 NTP Extension Field Format

In NTPv4, one or more extension fields can be inserted after the header and before the MAC, which is always present when an extension field is present. Other than defining the field format, this document makes no use of the field contents. An extension field contains a request or response message in the format shown in Figure 14.

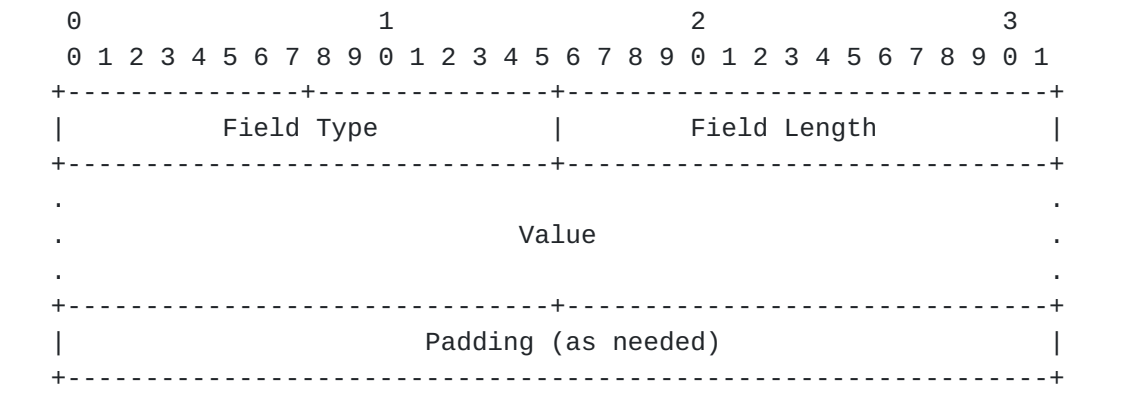


Figure 14: Extension Field Format

All extension fields are zero-padded to a word (four octets) boundary. The Field Type field is specific to the defined function and is not elaborated here. While the minimum field length containing required fields is four words (16 octets), a maximum field length remains to be established.

The Length field is a 16-bit unsigned integer that indicates the length of the entire extension field in octets, including the Padding field.

3.2. NEW: 7.5 NTP Extension Field Format

In NTPv4, one or more extension fields can be inserted after the header and before the possibly optional legacy MAC. A MAC SHOULD be present when an extension field is present. A MAC is always present in some form when NTP packets are authenticated. This MAC can be either a legacy MAC or a MAC-EF. Other than defining the field format, this document makes no use of the field contents. An extension field contains a request or response message in the format shown in Figure 14.

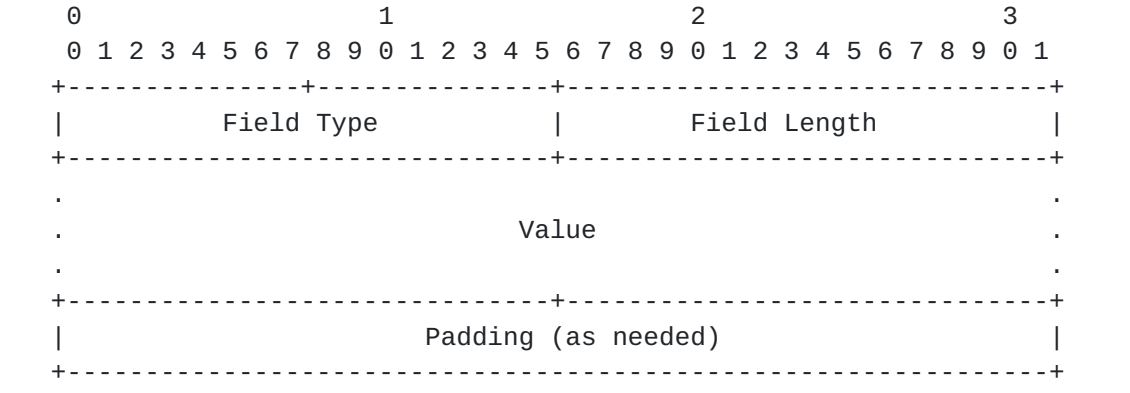
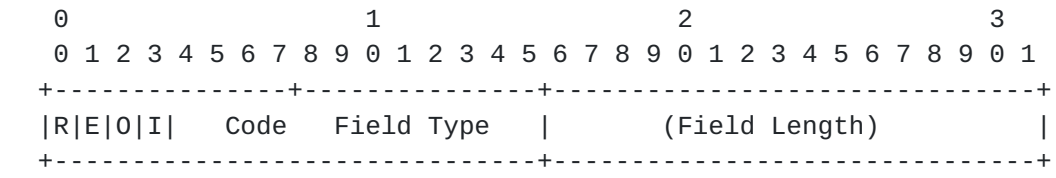


Figure 14: Extension Field Format

All extension fields are zero-padded to a word (four octets) boundary. The Field Type field is specific to the defined function and is not elaborated here. While the minimum field length containing required fields is four words (16 octets), a maximum field length remains to be established.

The Length field is a 16-bit unsigned integer that indicates the length of the entire extension field in octets, including the Padding field.

The Field Type contains the following sub-elements:



Field Type Format

Where:

R: 0 for a "Query", 1 for a "Response"

E: 0 for "OK", 1 for an "Error"

O: 0 for "MAC Required", 1 for "MAC Optional"

I: 0 for "MAC Not Included", 1 for "MAC Included"

The Code subtype is currently only used by RFC 5906, Autokey [RFC5906].

The Field Type, Value, and Padding fields are specific to the defined function and are not elaborated here; the Field Type value is defined in an IANA registry, and the Length, Value, and Padding values are defined by the document referred to by the registry. If a host receives an extension field with an unknown Field Type, the host SHOULD ignore the extension field and MAY drop the packet altogether if policy requires it.

While the minimum field length containing required fields is four words (16 octets), the maximum field length MUST NOT be longer than 65532 octets due to the maximum size of the data represented by the Length field, and SHOULD be small enough that the size of the NTP packet received by the client does not exceed the smallest MTU between the sender and the recipient.

The Length field is a 16-bit unsigned integer that indicates the length of the entire extension field in octets, including any Padding .

3.3. NEW: 7.5.1 Extension Fields and MACs

With the inclusion of additional Extension Fields, there is now a possibility of an ambiguous parsing of a legacy MAC. If an implementation offers even a modicum of care, there will be no ambiguity when parsing an NTP packet that contains a legacy MAC.

If an implementation uses the LAST-EF extension field, the presence of this field means "I am the last EF in this NTP Packet. Any subsequent packet data MUST be a legacy MAC." In this case, there is no parsing ambiguity.

If a system sends its MAC as a MAC-EF and does not send a legacy MAC, there is no parsing ambiguity.

The only time there is a potential for a parsing ambiguity is when a legacy MAC is provided and neither of the previous two cases are present. Even in this case, there is minimal risk.

An Extension Field contains a 2-octet Field Type, a 2-octet Field Length, and any payload (data and/or padding). If the NTP Packet parsing is at a point where it is evaluating data after the base packet, one of the following situations exists:

If the Field Length is not an even multiple of 4, we are not looking at an extension field. In this case, the only possibility of having a valid packet is if the data is part of a legacy MAC.

If the Field Length is valid, i.e., an even multiple of 4 octets, one of the following three cases must be present:

First, the Field Length will be less than the remaining data. This means subsequent data must parse as some number of Extension Fields, optionally follwed by a legacy MAC.

Second, the Field Length will exactly match the remaining data.

The third case is where the Field Length is longer than the remaining packet data. In this case, the current parse cannot be a valid extension field.

Semantic checking may also be done to validate a potential legacy MAC. A legacy MAC is a four-octet Key Identifier followed by a message digest. The usual message digest is 16 octets long but may

be another size, depending on the digest algorithm. In the Reference Implementation, a Key Identifier between 1 and 65535, inclusive, is a symmetric key, while a Key Identifier that is > 65535 is an Autokey RFC 5905 [RFC5905], or similar. If the receiving system does not recognize the Key Identifier, the data CANNOT be a valid legacy MAC. If the receiving system recognizes the Key Identifier, then it also has knowledge of the digest algorithm and can make sure the digest payload is the proper length. If this is not the case, then the data CANNOT be a valid legacy MAC.

It is trivial to parse the data after the base NTP packet and come up with a list of potential parsings. A local policy choice can specify the precedence of the parsing options in this case.

If none of the parsings validate, the packet fails authentication. An implementation has three local policy choices available if LAST-EF is not used and a legacy MAC may be provided. First, the implementation may specify EF-precedence. Second, the implementation may specify legacy-MAC-precedence. Finally, the implementation may specify "best fit" precedence. In this last case, the packet will meet one of the three following criteria: First, none of the parsings will match. Again, this is a case of failed authentication. Second, exactly one parsing will match and that parsing will be accepted. Third, multiple parsings will match, in which case the implementation may choose its behavior.

Additionally, most EFs will require a MAC. If there is a syntactically-valid parsing that does not include a MAC but previously scanned EFs require a MAC, then in a multiple-choice parsing scenario where one of the choices does not include a MAC the "no MAC provided" choice SHOULD be eliminated.

Note well that this rare situation can be completely avoided by using LAST-EF, or by indicating that no legacy MAC will be used.

4. IANA Considerations

This memo requests IANA to allocate NTP Extension Field Types 0x0007 (I-Do), 0x2007 (I-Do, MAC OPTIONAL), 0x8007 (I-Do Response), and 0xA007 (I-Do Response, MAC OPTIONAL)for this proposal.

5. Security Considerations

Additional information TBD

6. Normative References

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

[RFC5905] Mills, D., Martin, J., Ed., Burbank, J., and W. Kasch, "Network Time Protocol Version 4: Protocol and Algorithms Specification", RFC 5905, DOI 10.17487/RFC5905, June 2010, <http://www.rfc-editor.org/info/rfc5905>.

[RFC5906] Haberman, B., Ed. and D. Mills, "Network Time Protocol Version 4: Autokey Specification", RFC 5906, DOI 10.17487/RFC5906, June 2010, <http://www.rfc-editor.org/info/rfc5906>.

[RFC7821] Mizrahi, T., "UDP Checksum Complement in the Network Time Protocol (NTP)", RFC 7821, DOI 10.17487/RFC7821, March 2016, <http://www.rfc-editor.org/info/rfc7821>.

Authors' Addresses

Tal Mizrahi
Marvell
6 Hamda St.
Yokneam, 20692
Israel

Email: talmi@marvell.com

Danny Mayer
Network Time Foundation
P.O. Box 918
Talent, OR 97540
US

Email: mayer@ntp.org

Harlan Stenn
Network Time Foundation
P.O. Box 918
Talent, OR 97540
US

Email: stenn@nwtime.org