Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 10, 2019

G. Mirsky
ZTE Corp.
G. Jun
ZTE Corporation
H. Nydell
Accedian Networks
R. Foote
Nokia

September 6, 2018

Simple Two-way Active Measurement Protocol
draft-ietf-ippm-stamp-02

Abstract

This document describes a Simple Two-way Active Measurement Protocol which enables measurement of both one-way and round-trip performance metrics like delay, delay variation, and packet loss.

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 March 10, 2019.

Copyright Notice

Table of Contents

1. Introduction

Development and deployment of Two-Way Active Measurement Protocol (TWAMP) [RFC5357] and its extensions, e.g., [RFC6038] that defined features such as Reflect Octets and Symmetrical Size for TWAMP provided invaluable experience. Several independent implementations exist, have been deployed and provide important operational performance measurements. At the same time, there has been noticeable interest in using a simpler mechanism for active performance monitoring that can provide deterministic behavior and inherit separation of control (vendor-specific configuration or orchestration) and test functions. One of such is Performance Measurement from IP Edge to Customer Equipment using TWAMP Light from Broadband Forum ([BBF.TR-390]). This document defines active performance measurement test protocol, Simple Two-way Active Measurement Protocol (STAMP), that enables measurement of both one-way and round-trip performance metrics like delay, delay variation and packet loss.

2. Conventions used in this document

2.1. Terminology

STAMP - Simple Two-way Active Measurement Protocol

NTP - Network Time Protocol

PTP - Precision Time Protocol

HMAC Hashed Message Authentication Code

OWAMP One-Way Active Measurement Protocol

TWAMP Two-Way Active Measurement Protocol

2.2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Softwarization of Performance Measurement

Figure 1 presents Simple Two-way Active Measurement Protocol (STAMP) Session-Sender and Session-Reflector with a measurement session. The configuration and management of the STAMP Session-Sender, Session-Reflector and management of the STAMP sessions can be achieved through various means. Command Line Interface, OSS/BSS using SNMP or SDN using Netconf/YANG are but a few examples.

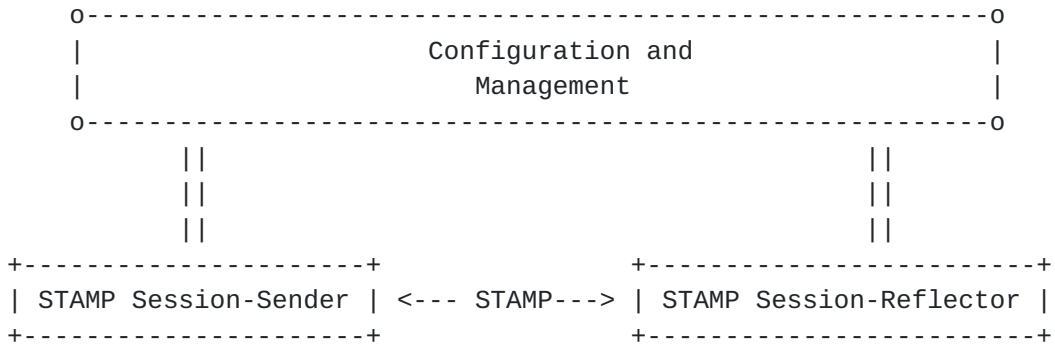


Figure 1: STAMP Reference Model

4. Theory of Operation

STAMP Session-Sender transmits test packets toward STAMP Session-Reflector. STAMP Session-Reflector receives Session-Sender's packet and acts according to the configuration and optional control information communicated in the Session-Sender's test packet. STAMP defines two different test packet formats, one for packets transmitted by the STAMP-Session-Sender and one for packets transmitted by the STAMP-Session-Reflector. STAMP supports three modes: unauthenticated, authenticated, and encrypted. Unauthenticated STAMP test packets are compatible on the wire with unauthenticated TWAMP-Test [RFC5357] packet formats.

By default, STAMP uses symmetrical packets, i.e., size of the packet transmitted by Session-Reflector equals the size of the packet received by the Session-Reflector.

4.1. Session-Sender Behavior and Packet Format

4.1.1. Session-Sender Packet Format in Unauthenticated Mode

Because STAMP supports symmetrical test packets, STAMP Session-Sender packet has a minimum size of 44 octets in unauthenticated mode, see Figure 2, and 48 octets in authenticated or encrypted modes, see Figure 4.

For unauthenticated mode:

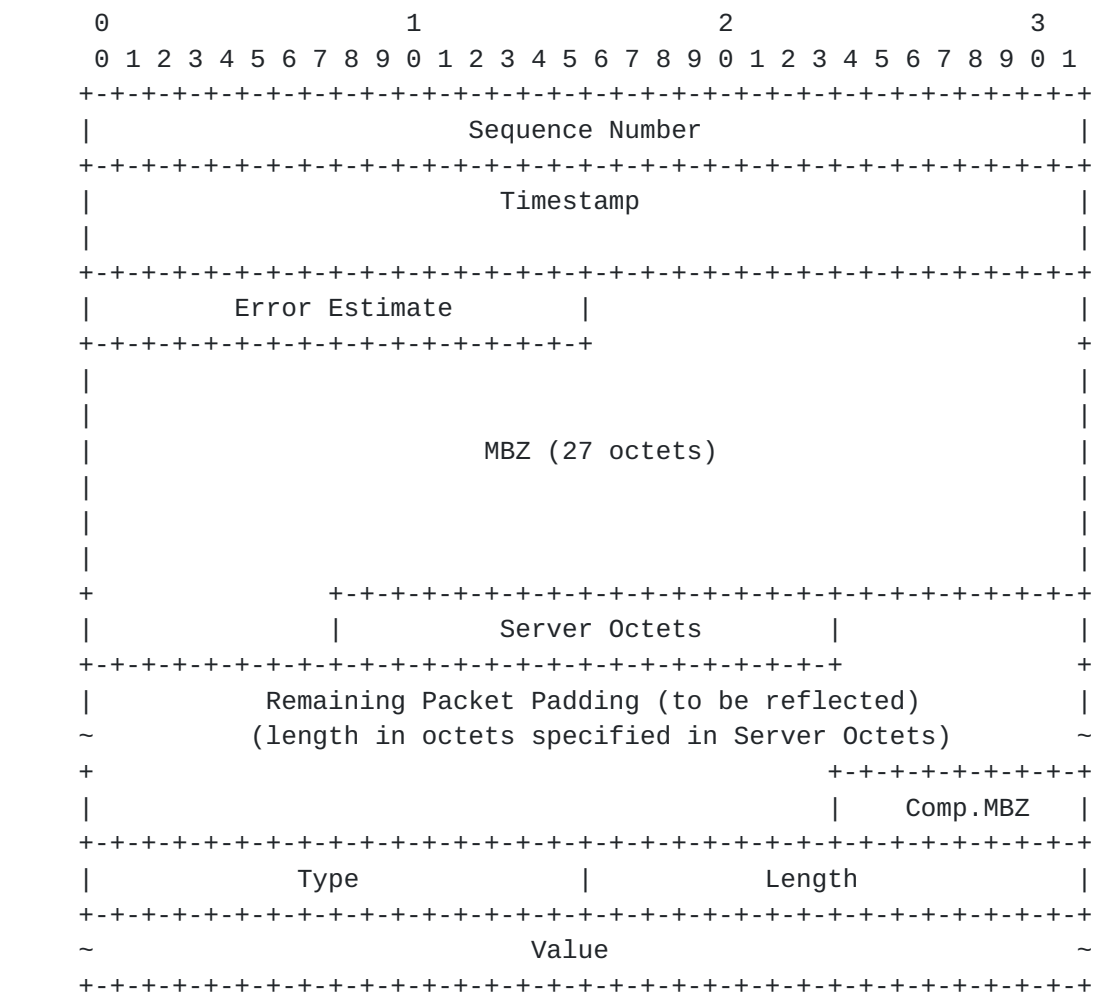


Figure 2: STAMP Session-Sender test packet format in unauthenticated mode

where fields are defined as the following:

- Sequence Number is four octets long field. For each new session its value starts at zero and is incremented with each transmitted packet.

- Timestamp is eight octets long field. STAMP node MUST support Network Time Protocol (NTP) version 4 64-bit timestamp format [RFC5905]. STAMP node MAY support IEEE 1588v2 Precision Time Protocol truncated 64-bit timestamp format [IEEE.1588.2008].

- Error Estimate is two octets long field with format displayed in Figure 3

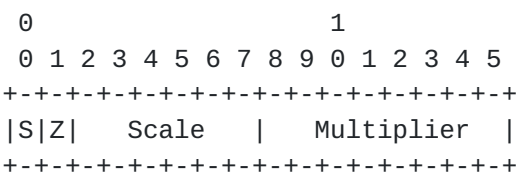


Figure 3: Error Estimate Format

where S, Scale, and Multiplier fields are interpreted as they have been defined in section 4.1.2 [RFC4656]; and Z field - as has been defined in section 2.3 [RFC8186]:

* 0 - NTP 64 bit format of a timestamp;

* 1 - PTPv2 truncated format of a timestamp.

The STAMP Session-Sender and Session-Reflector MAY use, not use, or set value of the Z field in accordance with the timestamp format in use. This optional field is to enhance operations, but local configuration or defaults could be used in its place.

- Must-be-Zero (MBZ) field in the session-sender unauthenticated packet is 27 octets long. It MUST be all zeroed on the transmission and ignored on receipt.

- Server Octets field is two octets long field. It MUST follow the 27 octets long MBZ field. The Reflect Octets capability defined in [RFC6038]. The value in the Server Octets field equals to the number of octets the Session-Reflector is expected to copy back to the Session-Sender starting with the Server Octets field. Thus the minimal non-zero value for the Server Octets field is two. Therefore, the value of one is invalid. If none of Payload to be copied, the value of the Server Octets field MUST be set to zero on transmit.

- Remaining Packet Padding is an optional field of variable length. The number of octets in the Remaining Packet Padding field is the value of the Server Octets field less the length of the Server Octets field.

- Comp.MBZ is variable length field used to achieve alignment on a word boundary. Thus the length of Comp.MBZ field may be only 0, 1, 2 or 3 octets. The value of the field MUST be zeroed on transmission and ignored on receipt.

The unauthenticated STAMP Session-Sender packet MAY include Type-Length-Value encodings that immediately follow the Comp. MBZ field.

- Type field is two octets long. The value of the Type field is the codepoint allocated by IANA Section 5 that identifies data in the Value field.

- Length is two octets long field, and its value is the length of the Value field in octets.

- Value field contains the application specific information. The length of the Value field MUST be four octets aligned.

4.1.2. Session-Sender Packet Format in Authenticated and Encrypted Modes

For authenticated and encrypted modes:

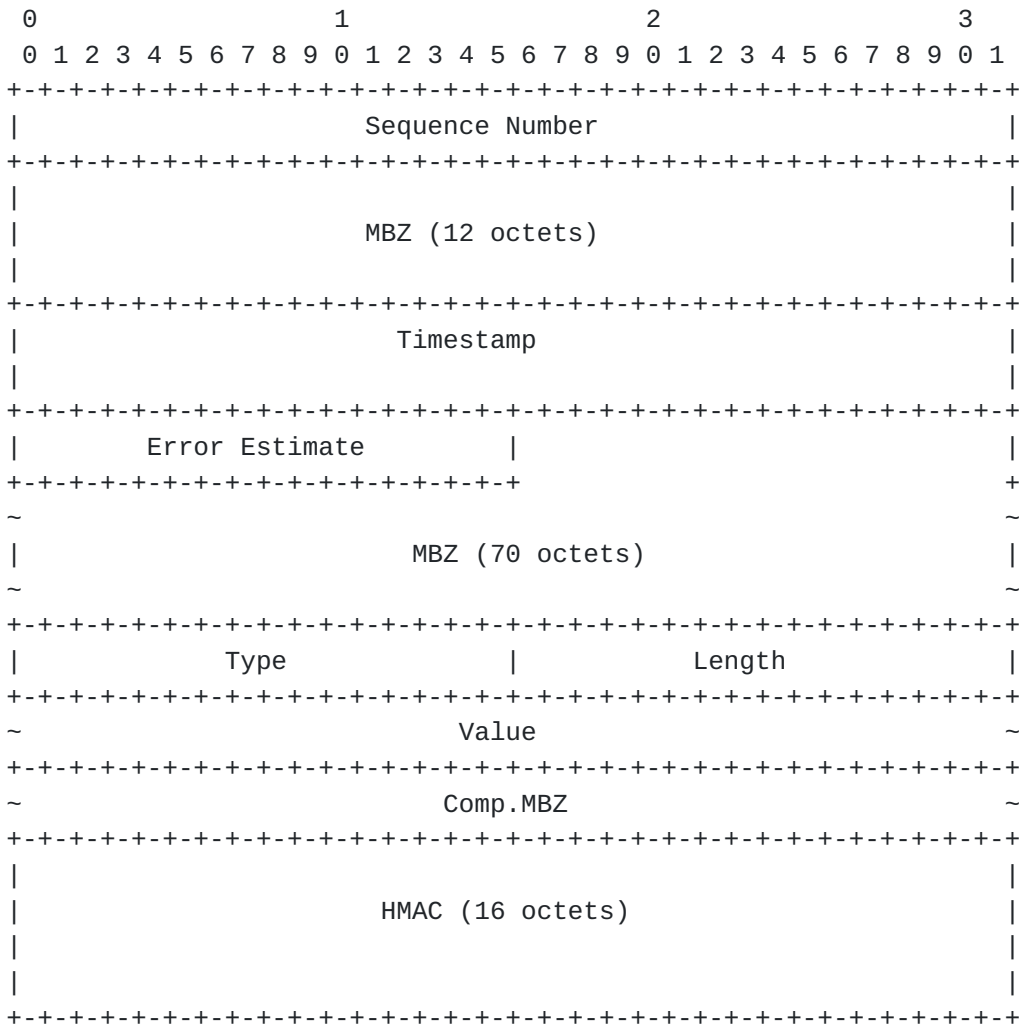


Figure 4: STAMP Session-Sender test packet format in authenticated or encrypted modes

The field definitions are the same as the unauthenticated mode, listed in Section 4.1.1. Also, Comp.MBZ field is variable length filed to align the packet on 16 octets boundary. Also, the packet includes a key-hashed message authentication code (HMAC) ([RFC2104]) hash at the end of the PDU.

The STAMP Session-Sender-packet format (Figure 4) is the same in authenticated and encrypted modes. The encryption and authentication operations are, however, different and protect the data as follows:

in the authenticated mode the Sequence Number is protected while the Timestamp and the Error Estimate are sent in clear text;

in encrypted mode all fields, including the timestamp and Error Estimate, are protected to provide maximum data confidentiality and integrity protection.

Sending the Timestamp in clear text in authenticated mode allows more consistent reading of time by a Session-Sender on the transmission of the test packet. Reading of the time in encrypted mode must be followed by its encryption which introduces variable delay thus affecting calculated timing metrics.

4.2. Session-Reflector Behavior and Packet Format

The Session-Reflector receives the STAMP test packet, verifies it, prepares and transmits the reflected test packet.

Two modes of STAMP Session-Reflector characterize the expected behavior and, consequently, performance metrics that can be measured:

- Stateless - STAMP Session-Reflector does not maintain test state and will reflect the received sequence number without modification. As a result, only round-trip packet loss can be calculated while the reflector is operating in stateless mode.

- Stateful - STAMP Session-Reflector maintains test state thus enabling the ability to determine forward loss, gaps recognized in the received sequence number. As a result, both near-end (forward) and far-end (backward) packet loss can be computed. This implies that the STAMP Session-Reflector MUST keep a state for each accepted STAMP-test session, uniquely identifying STAMP-test packets to one such session instance, and enabling adding a sequence number in the test reply that is individually incremented on a per-session basis.

4.2.1. Session-Reflector Packet Format in Unauthenticated Mode

For unauthenticated mode:

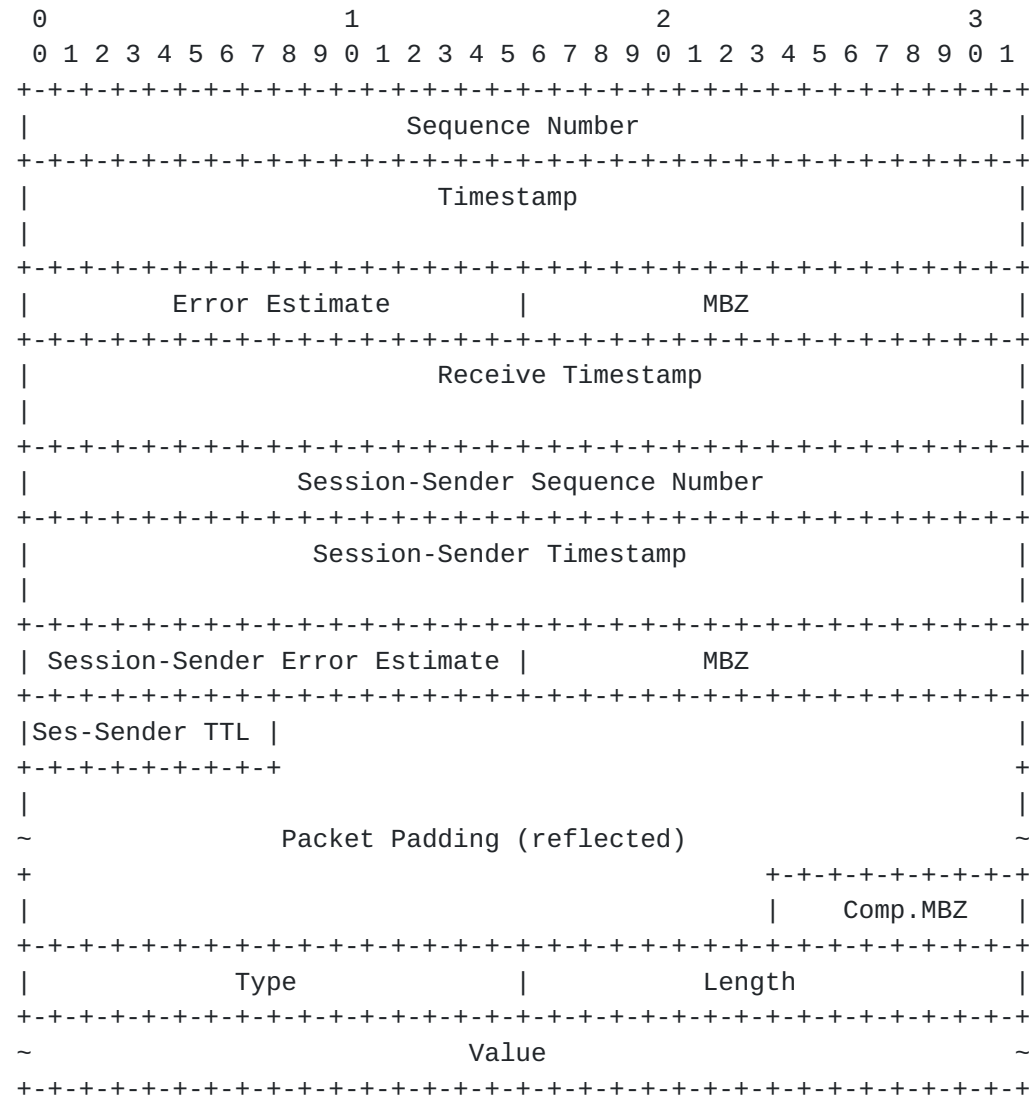


Figure 5: STAMP Session-Reflector test packet format in unauthenticated mode

where fields are defined as the following:

o Sequence Number is four octets long field. The value of the Sequence Number field is set according to the mode of the STAMP Session-Reflector:

  * in the stateless mode the Session-Reflector copies the value from the received STAMP test packet's Sequence Number field;

  * in the stateful mode the Session-Reflector counts the received STAMP test packets in each test session and uses that counter to set the value of the Sequence Number field.

- Timestamp and Receiver Timestamp fields are each eight octets long. The format of these fields, NTP or PTPv2, indicated by the Z flag of the Error Estimate field as described in Section 4.1.

- Error Estimate has the same size and interpretation as described in Section 4.1.

- Session-Sender Sequence Number, Session-Sender Timestamp, and Session-Sender Error Estimate are copies of the corresponding fields in the STAMP test packet sent by the Session-Sender.

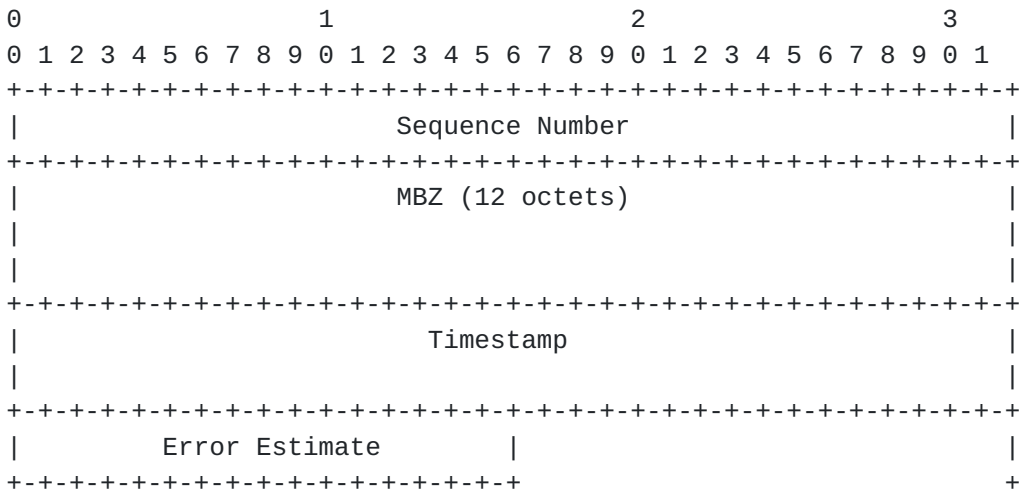
- Ses(sion)-Sender TTL is one octet long field, and its value is the copy of the TTL field from the received STAMP test packet.

- Packet Padding (reflected) is optional variable length field. The length of the Packet Padding (reflected) field MUST be equal to the value of the Server Octets field (Figure 2). If the value is non-zero, the Session-Reflector copies octets starting with the Server Octets field.

- Comp.MBZ is variable length field used to achieve alignment on a word boundary. Thus the length of Comp.MBZ field may be only 0, 1, 2 or 3 octets. The value of the field MUST be zeroed on transmission and ignored on receipt.

4.2.2. Session-Reflector Packet Format in Authenticated and Encrypted Modes

For authenticated and encrypted modes:



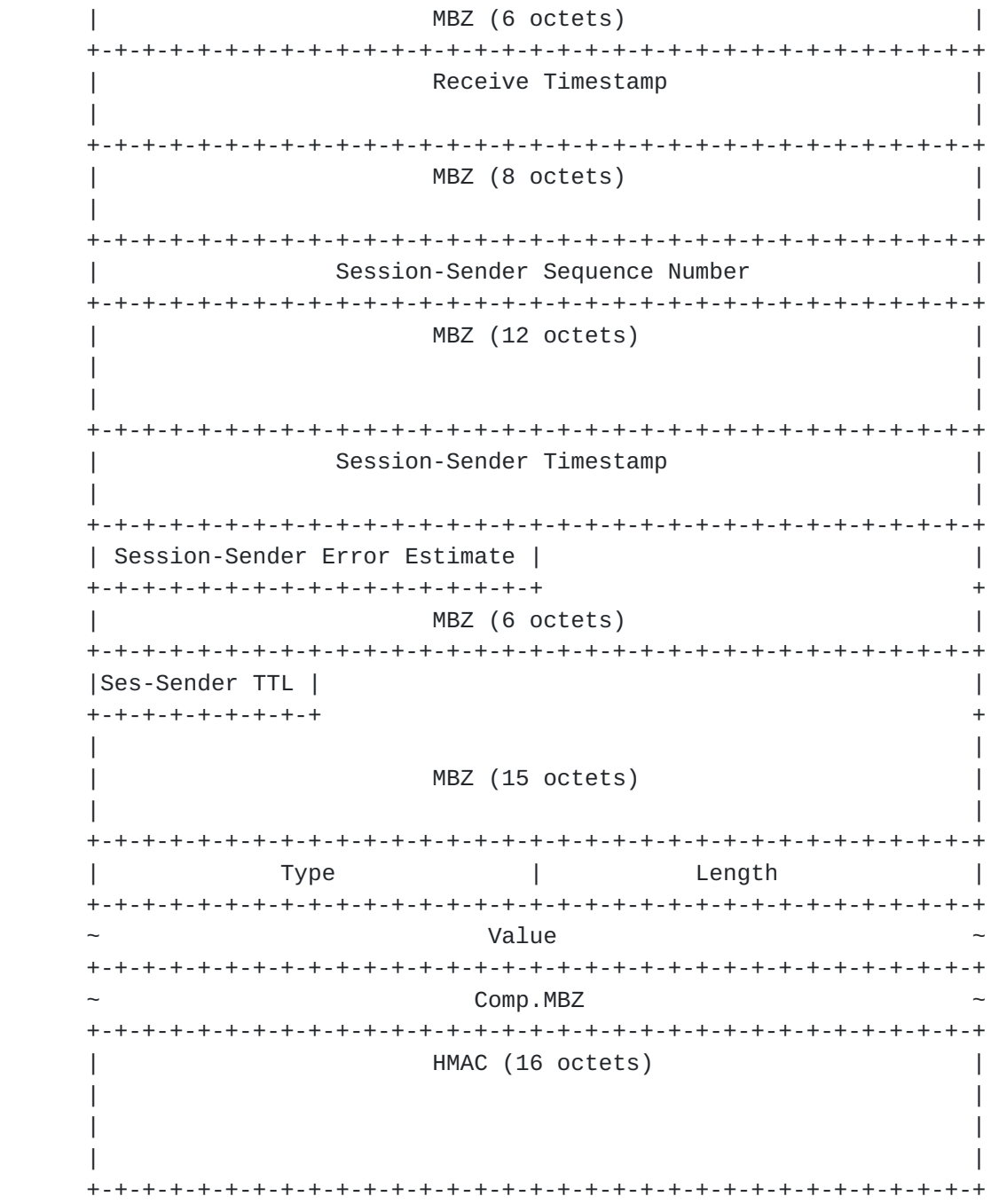


Figure 6: STAMP Session-Reflector test packet format in authenticated or encrypted modes

The field definitions are the same as the unauthenticated mode, listed in Section 4.2.1, and includes a key-hashed message authentication code (HMAC) ([RFC2104]) hash at the end of the PDU.

4.3. Interoperability with TWAMP Light

One of the important requirements to STAMP is the ability to interwork with TWAMP Light device. There are two possible combinations for such use case:

- STAMP Session-Sender with TWAMP Light Session-Reflector;
- TWAMP Light Session-Sender with STAMP Session-Reflector.

In the former case, Session-Sender MAY not be aware that its Session-Reflector does not support STAMP. For example, TWAMP Light Session-Reflector may not support the use of UDP port 862 as defined in [I-D.ietf-ippm-port-twamp-test]. But because STAMP Session-Sender MUST be able to send test packets to destination UDP port number from the Dynamic and/or Private Ports range 49152-65535, test management system should find port number that both devices can use. And if any of TLV-based STAMP extensions are used, the TWAMP Light Session-Reflector will view them as Packet Padding field. The Session-Sender SHOULD use the default format for its timestamps - NTP. And it MAY use PTPv2 timestamp format.

In the latter scenario, the test management system should set STAMP Session-Reflector to use UDP port number from the Dynamic and/or Private Ports range. As for Packet Padding field that the TWAMP Light Session-Sender includes in its transmitted packet, the STAMP Session-Reflector will process it according to [RFC6038] and return reflected packet of the symmetrical size. The Session-Reflector MUST use the default format for its timestamps - NTP.

5. IANA Considerations

This document doesn't have any IANA action. This section may be removed before the publication.

6. Security Considerations

Use of HMAC in authenticated and encrypted modes may be used to simultaneously verify both the data integrity and the authentication of the STAMP test packets.

7. Acknowledgments

TBD

8. References

8.1. Normative References

[BBF.TR-390]
"Performance Measurement from IP Edge to Customer Equipment using TWAMP Light", BBF TR-390, May 2017.

[I-D.ietf-ippm-port-twamp-test]
Morton, A. and G. Mirsky, "OWAMP and TWAMP Well-Known Port Assignments", draft-ietf-ippm-port-twamp-test-01 (work in progress), March 2018.

[IEEE.1588.2008]
"Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems", IEEE Standard 1588, March 2008.

[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>.

[RFC4656] Shalunov, S., Teitelbaum, B., Karp, A., Boote, J., and M. Zekauskas, "A One-way Active Measurement Protocol (OWAMP)", RFC 4656, DOI 10.17487/RFC4656, September 2006, <https://www.rfc-editor.org/info/rfc4656>.

[RFC5357] Hedayat, K., Krzanowski, R., Morton, A., Yum, K., and J. Babiarz, "A Two-Way Active Measurement Protocol (TWAMP)", RFC 5357, DOI 10.17487/RFC5357, October 2008, <https://www.rfc-editor.org/info/rfc5357>.

[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, <https://www.rfc-editor.org/info/rfc5905>.

[RFC6038] Morton, A. and L. Ciavattone, "Two-Way Active Measurement Protocol (TWAMP) Reflect Octets and Symmetrical Size Features", RFC 6038, DOI 10.17487/RFC6038, October 2010, <https://www.rfc-editor.org/info/rfc6038>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8186] Mirsky, G. and I. Meilik, "Support of the IEEE 1588 Timestamp Format in a Two-Way Active Measurement Protocol (TWAMP)", RFC 8186, DOI 10.17487/RFC8186, June 2017, <https://www.rfc-editor.org/info/rfc8186>.

8.2. Informative References

[RFC2104] Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", RFC 2104, DOI 10.17487/RFC2104, February 1997, <https://www.rfc-editor.org/info/rfc2104>.

Authors' Addresses

Greg Mirsky
ZTE Corp.

Email: gregimirsky@gmail.com

Guo Jun
ZTE Corporation
68# Zijinghua Road
Nanjing, Jiangsu 210012
P.R.China

Phone: +86 18105183663
Email: guo.jun2@zte.com.cn

Henrik Nydell
Accedian Networks

Email: hnydell@accedian.com

Richard Foote
Nokia

Email: footer.foote@nokia.com