

IPPM Working Group
Internet-Draft
Intended status: Standards Track
Expires: 19 December 2025

R. Gandhi, Ed.
Cisco Systems, Inc.
P. Schoenmaker
Meta Platforms, Inc.
17 June 2025

Simple Two-Way Active Measurement Protocol (STAMP) Extensions for Bit
Error Rate Measurement
draft-gandhi-ippm-stamp-ber-02

Abstract

The Simple Two-Way Active Measurement Protocol (STAMP), as defined in RFC 8762, along with its optional extensions specified in RFC 8972, can be utilized for active measurement. This document further augments the STAMP extensions specified in RFC 8972 to enable the measurement of the bit error rate within the "Extra Padding" TLV of STAMP packets.

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 19 December 2025.

Copyright Notice

Copyright (c) 2025 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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	2
2. Conventions Used in This Document	3
2.1. Requirements Language	3
2.2. Abbreviations	3
2.3. STAMP Reference Topology	4
3. Overview	4
3.1. Bit Errors in Non-measurement Fields of STAMP	5
4. STAMP Procedure	5
4.1. STAMP Session-Sender	6
4.1.1. Considerations for Bit Pattern	7
4.2. STAMP Session-Reflector	7
4.2.1. STAMP TLV Conformant Check	7
4.3. Considerations for Link Aggregation Group	8
5. STAMP Extensions	8
5.1. Bit Pattern in Padding STAMP TLV	8
5.2. Bit Error Count in Padding STAMP TLV	9
6. Data Model Parameters	9
6.1. Configuration Data Model Parameters	9
6.2. Operational Data Model Parameters	10
7. Security Considerations	10
8. Implementation Status	10
9. IANA Considerations	11
10. References	11
10.1. Normative References	11
10.2. Informative References	12
Acknowledgments	12
Authors' Addresses	12

1. Introduction

The Simple Two-Way Active Measurement Protocol (STAMP) is designed to measure various performance metrics in IP networks without relying on a control channel to pre-signal session parameters, as specified in [RFC8762]. STAMP test packets are sent between a Session-Sender and a Session-Reflector to measure delay and packet loss along the path.

[RFC8972] introduces optional extensions for STAMP in the form of Type-Length-Value (TLV) objects, including the capability to transmit "Extra Padding" TLV within STAMP test packets.

Networks may experience transmission bit errors due to various factors, such as poor fiber quality. The bit error can be a single bit error or a burst of bit errors at a time. It is beneficial to measure the Bit Error Rate (BER) using active measurement packets between two nodes. For accurate BER measurement, transmitting large-sized active measurement packets is preferable, especially on links with low bit error rates. Furthermore, there is a need to transmit test packets at a high rate to measure BER on high-capacity links.

The STAMP test packets use a UDP header with a checksum field that may be used for checking the integrity of the header and data. The UDP checksum is optional for the IPv4 header and may be set to 0 for the IPv6 header for the STAMP destination UDP port. However, the checksum field does not provide an accurate measurement of bit errors.

Authenticated mode provides data integrity protection for the STAMP test packets by adding a Hashed Message Authentication Code (HMAC), such as HMAC-SHA-256 [RFC8762]. However, the authenticated mode does not provide an accurate measurement of bit errors. In addition, the HMAC TLV defined in [RFC8972] for authenticating STAMP TLVs does not include checking the "Extra Padding" TLV.

This document further augments the STAMP extensions defined in [RFC8972] to enable the measurement of BER within the "Extra Padding" TLV of STAMP packets.

2. Conventions Used in This Document

2.1. 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.

2.2. Abbreviations

BER: Bit Error Rate

MTU: Maximum Transmission Unit

STAMP: Simple Two-way Active Measurement Protocol

TLV: Type-Length-Value

2.3. STAMP Reference Topology

In the STAMP reference topology shown in Figure 1, the STAMP Session-Sender S1 initiates Session-Sender test packets, and the STAMP Session-Reflector R1 transmits reply Session-Reflector test packets.

T1 is a transmit timestamp, and T4 is a receive timestamp added by node S1. T2 is a receive timestamp, and T3 is a transmit timestamp added by node R1.

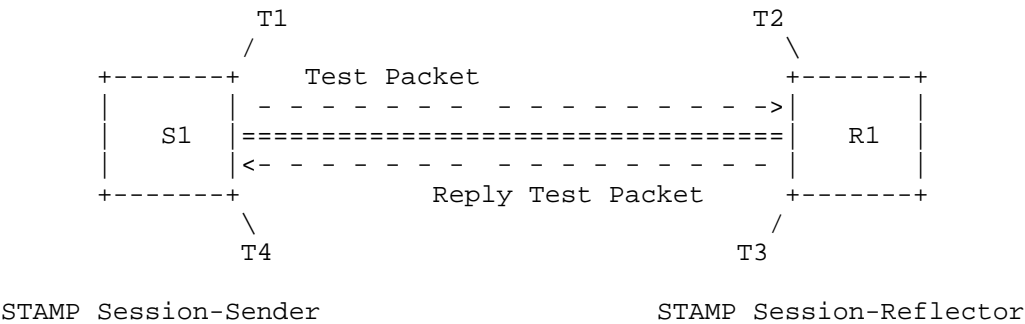


Figure 1: STAMP Reference Topology

3. Overview

The optional extensions for STAMP test packets [RFC8762] are defined in [RFC8762] in the form of TLVs. The Session-Sender transmits optional STAMP TLVs, and the Session-Reflector reflects all received STAMP TLVs from the Session-Sender test packets. [RFC8972] defines an optional TLV extension specifically for transmitting "Extra Padding" (Type=1) TLV in the STAMP test packets. The "Extra Padding" TLV can be filled using either a predefined fixed pattern or a random pattern of bits [RFC8972].

This document defines a procedure to measure BER within the "Extra Padding" TLV. The process involves the Session-Sender transmitting the extra padding filled with a predefined bit pattern. The Session-Reflector then checks for bit errors by comparing the received padding against the predefined bit pattern. This allows for the detection of a single bit error or a burst of bit errors and the measurement of the BER. The Session-Reflector does not discard the STAMP test packet with bit errors but instead reflects it back to the Session-Sender after correcting the bit errors. The Session-Reflector also returns the bit error count to the Session-Sender.

BER is measured in both the forward and reverse directions between the Session-Sender and the Session-Reflector using the procedure and extensions defined in this document. The BER is calculated using the number of bit errors detected and the number of bits received in the extra padding.

As specified in [RFC8972], the Session-Sender and Session-Reflector test packets are symmetric in size. The Session-Sender and Session-Reflector MUST ensure that the resulting test packets do not exceed the path MTU after adding the STAMP TLVs.

3.1. Bit Errors in Non-measurement Fields of STAMP

Note that the procedure and extensions defined in this document do not use the base STAMP packets, packet headers, or STAMP TLVs other than the "Extra Padding" TLV for BER measurement. It is possible that the bit errors impact those non-measurement fields of the STAMP test packets causing verification failures. Such STAMP test packets are reported using a different measurement metric. The integrity of those fields can be verified using the HMAC mechanisms defined in [RFC8762] and [RFC8972].

4. STAMP Procedure

This document defines two TLV options for STAMP: "Bit Pattern in Padding" TLV (Type=TBA1) and "Bit Error Count in Padding" TLV (Type=TBA2).

An example of a STAMP test packet used for measuring BER is shown in Figure 2. It uses the "Extra Padding" TLV, the optional "Bit Pattern in Padding" TLV, and the "Bit Error Count in Padding" TLV.

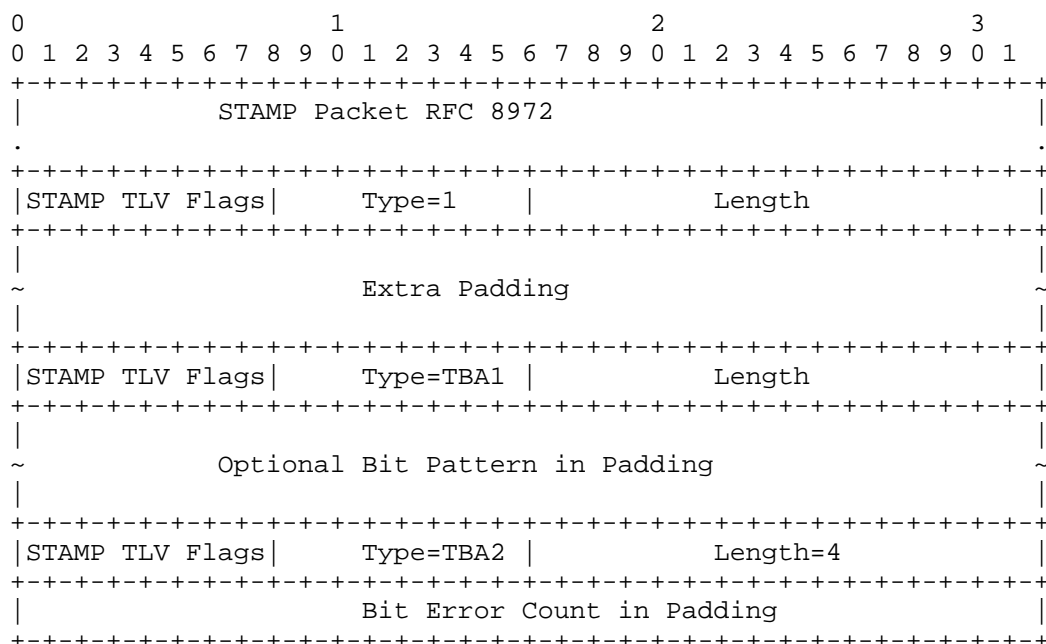


Figure 2: Example STAMP Packet to Measure BER

4.1. STAMP Session-Sender

When a STAMP Session-Sender is set up to measure BER, it adds an "Extra Padding" (Type=1) TLV, a "Bit Error Count in Padding" (Type=TBA2) TLV, and optionally, a "Bit Pattern in Padding" (Type=TBA1) TLV in Session-Sender test packets. The Session-Sender test packets carry only one "Bit Error Count in Padding" TLV, only one "Extra Padding" TLV [RFC8972] and optionally carry only one "Bit Pattern in Padding" TLV.

The Session-Sender MUST add an "Extra Padding" TLV [RFC8972] when it adds a "Bit Pattern in Padding" TLV to the Session-Sender test packets. The variable-length data in the "Bit Pattern in Padding" TLV MUST contain the bit pattern employed in the "Extra Padding" TLV. It is RECOMMENDED to have the length of the extra padding as an integer multiple of the length of the Bit Pattern to ease implementation.

The Session-Sender MUST also add an "Extra Padding" TLV [RFC8972] when it adds a "Bit Error Count in Padding" TLV in the Session-Sender test packets. The bit error count in padding MUST be set to 0.

Note that the integrity of the "Bit Pattern in Padding" and "Bit Error Count in Padding" TLVs can be protected using the HMAC mechanisms defined in [RFC8972].

4.1.1. Considerations for Bit Pattern

It is possible that the bit pattern in the "Bit Pattern in Padding" TLV itself has bit errors. This can result in a measurement error due to mismatch between the bit pattern and the extra padding. One way to avoid this issue is for the Session-Sender and Session-Reflector to use the local configuration with the default value of 0xFF00 as the bit pattern. In this case, the "Bit Pattern in Padding" TLV is not transmitted in the STAMP test packets.

4.2. STAMP Session-Reflector

When the Session-Reflector receives a STAMP test packet with a "Bit Pattern in Padding" TLV, the Session-Reflector that supports this TLV MUST check the extra padding in the "Extra Padding" TLV against the bit pattern to detect any bits that do not match the bit pattern and count them as bit errors.

When the Session-Reflector receives a STAMP test packet with a "Bit Error Count in Padding" TLV, the Session-Reflector that supports this TLV MUST check the "Extra Padding" TLV against the expected bit pattern to detect if there are any bits not matching the bit pattern and count them as bit errors. The Session-Reflector updates the count of bit errors in the received "Bit Error Count in Padding" TLV and reflects the TLV back to the Session-Sender. If no bit errors are detected, the bit error count remains as 0 in the reflected "Bit Error Count in Padding" TLV.

The Session-Reflector corrects the bit errors in the "Extra Padding" TLV by matching the bit pattern and reflects the corrected "Extra Padding" TLV to the Session-Sender. The corrected "Extra Padding" TLV is used to measure the BER in the reverse direction.

4.2.1. STAMP TLV Conformance Check

If the Session-Reflector receives a STAMP test packet with a "Bit Pattern in Padding" TLV or a "Bit Error Count in Padding" TLV without an "Extra Padding" TLV or with more than one "Extra Padding" TLV, it MUST set the C flag (Conformant) defined in [I-D.ietf-ippm-asymmetrical-pkts] to 1 in the STAMP TLV Flags in the reflected STAMP test packet for those STAMP TLVs.

If the Session-Reflector receives a STAMP test packet that contains more than one "Bit Pattern in Padding" TLV or more than one "Bit Error Count in Padding" TLV, it MUST set the C flag (Conformant) defined in [I-D.ietf-ippm-asymmetrical-pkts] to 1 in the STAMP TLV Flags in the reflected STAMP test packet for those STAMP TLVs.

4.3. Considerations for Link Aggregation Group

Networks may experience transmission bit errors differently for different link members of a Link Aggregation Group (LAG). The procedure and extensions defined in this document are equally applicable for measuring BER for each individual member of the LAG.

A separate STAMP micro-session is created for each member of the LAG, as defined in [RFC9534]. The STAMP extension for the Micro-Session ID TLV, as defined in [RFC9534], is used to identify each member link of the LAG associated with the STAMP micro-session on the Session-Sender and Session-Reflector. The Session-Reflector replies on the same member of the LAG in the reverse direction based on the received Session-Sender test packets and the local configuration or the received information from the data plane.

Note that in order to get a good approximation of the BER, it is RECOMMENDED to transmit the STAMP test packets that match the link MTU size.

5. STAMP Extensions

5.1. Bit Pattern in Padding STAMP TLV

The "Bit Pattern in Padding" TLV is optional and is carried by Session-Sender and Session-Reflector test packets. The format of the TLV is shown in Figure 3.

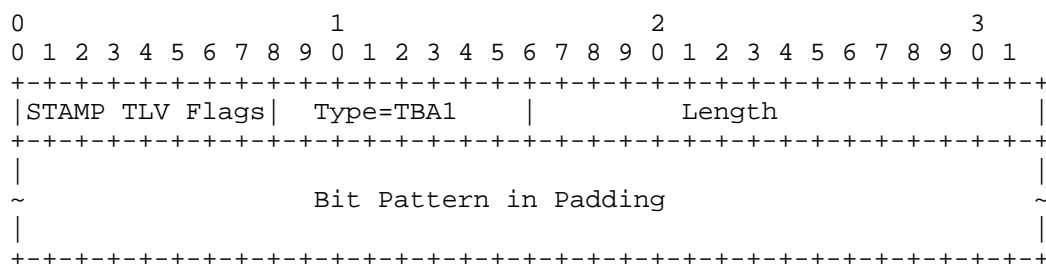


Figure 3: Bit Pattern in Padding STAMP TLV

The TLV fields are defined as follows:

Type: Type (value TBA1)

STAMP TLV Flags: The STAMP TLV Flags follow the procedures described in [RFC8972].

Length: A two-octet field equal to the length of the Data in octets.

5.2. Bit Error Count in Padding STAMP TLV

The "Bit Error Count in Padding" TLV is optional and is carried by Session-Sender and Session-Reflector test packets. The format of the TLV is shown in Figure 4.

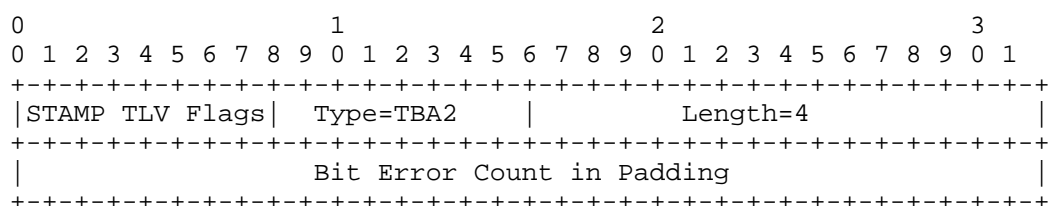


Figure 4: Bit Error Count in Padding STAMP TLV

The TLV fields are defined as follows:

Type: Type (value TBA2)

STAMP TLV Flags: The STAMP TLV Flags follow the procedures described in [RFC8972].

Length: A two-octet field set to 4 for the Data.

6. Data Model Parameters

6.1. Configuration Data Model Parameters

The configuration data model for the BER measurement using STAMP MUST allow to set the following parameters:

- Padding size (number of bytes)
- Padding bit pattern (with variable length of bytes)
- Transmit interval for STAMP test packets
- Computation interval as a multiple of transmit interval for reporting the BER

6.2. Operational Data Model Parameters

The operational data model for the BER measurement using STAMP MUST allow to telemetry the following parameters:

Forward direction BER measurement:

- Number of total packets received in the computation interval
- Number of total packets received with non-zero Bit Error Count in TLV in the computation interval
- Number of total bits in the padding TLV of all received packets in the computation interval
- Number of total Bit Error Count in TLV of all received packets in the computation interval

Reverse direction BER measurement:

- Number of total packets received in the computation interval
- Number of total packets received with bit errors in the computation interval
- Number of total bits in the padding TLV of all received packets in the computation interval
- Number of total bit errors in all received packets in the computation interval

Thresholds are defined for the forward and reverse directions of the BER measurement, as number of bit errors per million and number of packets with bit errors per million, computed during the computation interval. An alarm is generated, and an event-driven telemetry is triggered when the computed metric crosses the threshold.

7. Security Considerations

The security considerations specified in [RFC8762] and [RFC8972] apply to the procedure and extensions defined in this document.

8. Implementation Status

Editorial note: Please remove this section prior to publication.

An open-source implementation of the Simple Two-Way Active Measurement Protocol (RFC 8762) is available in Teaparty.

<https://github.com/cerfcast/teaparty>

An implementation of the solution in this document is available at the following location:

<https://github.com/cerfcast/teaparty/commit/592558a38dbcf9b273acb2a2fe8ab0d8f16d0709>

And (as bonus) there is also support for the BER in the Wireshark dissector:

<https://github.com/cerfcast/teaparty/commit/608b9e89fce2f25ed88eaa367d0bacc693845da2>

Contact:

William Hawkins

University of Cincinnati

Email: hawkinsw@obs.cr

9. IANA Considerations

IANA has created the "STAMP TLV Types" registry for [RFC8972]. IANA is requested to allocate a value for the "Bit Pattern in Padding" TLV Type and a value for the "Bit Error Count in Padding" TLV Type from the IETF Review TLV range of the same registry.

Value	Description	Reference
TBA1	Bit Pattern in Padding	This document
TBA2	Bit Error Count in Padding	This document

Table 1: STAMP TLV Types

10. References

10.1. 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, <<https://www.rfc-editor.org/info/rfc2119>>.

- [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>>.
- [RFC8762] Mirsky, G., Jun, G., Nydell, H., and R. Foote, "Simple Two-Way Active Measurement Protocol", RFC 8762, DOI 10.17487/RFC8762, March 2020, <<https://www.rfc-editor.org/info/rfc8762>>.
- [RFC8972] Mirsky, G., Min, X., Nydell, H., Foote, R., Masputra, A., and E. Ruffini, "Simple Two-Way Active Measurement Protocol Optional Extensions", RFC 8972, DOI 10.17487/RFC8972, January 2021, <<https://www.rfc-editor.org/info/rfc8972>>.
- [I-D.ietf-ippm-asymmetrical-pkts]
Mirsky, G., Ruffini, E., Nydell, H., Foote, R. F., and W. Hawkins, "Performance Measurement with Asymmetrical Traffic Using STAMP", Work in Progress, Internet-Draft, draft-ietf-ippm-asymmetrical-pkts-07, 5 May 2025, <<https://datatracker.ietf.org/doc/html/draft-ietf-ippm-asymmetrical-pkts-07>>.

10.2. Informative References

- [RFC9534] Li, Z., Zhou, T., Guo, J., Mirsky, G., and R. Gandhi, "Simple Two-Way Active Measurement Protocol Extensions for Performance Measurement on a Link Aggregation Group", RFC 9534, DOI 10.17487/RFC9534, January 2024, <<https://www.rfc-editor.org/info/rfc9534>>.

Acknowledgments

The authors would like to thank Ianik Semco and Miloslav Kopka for the discussions on the bit error rate measurements. The authors would also like to thank Ruediger Geib for reviewing this document and providing many useful comments and suggestions. The authors would also like to thank William Hawkins for implementing the solution defined in this document and providing many useful suggestions.

Authors' Addresses

Rakesh Gandhi (editor)
Cisco Systems, Inc.
Canada
Email: rgandhi@cisco.com

Peter Schoenmaker
Meta Platforms, Inc.
United Kingdom
Email: psch@meta.com