NTP
Internet-Draft
Intended status: Informational
Expires: 4 May 2026
S. Tang
G. Zhuang
Huawei Technologies
31 October 2025

NTPv5 Timestamp Extension Field
draft-tang-ntp-ntpv5-extension-field-00

Abstract

This document describes an alternative timestamp to the Monotonic Receive Timestamp Extension Field defined in NTP version 5 (NTPv5) when transferring frequency offset. The new extension field, named Monotonic RAW Receive Timestamp Extension Field uses a stable clock source that is not affected by NTP adjustment. It provides more accurate frequency-transfer offset between a remote server and local client, which further enhances the accuracy of time synchronization.

About This Document

This note is to be removed before publishing as an RFC.

The latest revision of this draft can be found at https://example.com/LATEST. Status information for this document may be found at https://datatracker.ietf.org/doc/draft-tang-ntp-ntpv5-extension-field/.

Discussion of this document takes place on the WG Working Group mailing list (mailto:WG@example.com), which is archived at https://example.com/WG.

Source for this draft and an issue tracker can be found at https://github.com/USER/REPO.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

NTP version 5 (NTPv5) [I-D.draft-ietf-ntp-ntpv5] introduces a Monotonic Receive Timestamp Extension Field to transfer frequency in addition to the time-transfer offset captured by the receive and transmit timestamps in the header. Separation of time and frequency transfer using different clocks shall enhance synchronization accuracy. It should be noted that when the system clock is slewed [RFC5905], the clock rate of the Monotonic Receive Timestamp Extension Field changes accordingly, i.e., clock rate diverges from the rate of the crystal. This introduces additional errors when performing frequency transfer, hence, negatively impact the accuracy of clock synchronization. This document proposes a stable clock source whose rate is not affected by NTP adjustment. The Monotonic RAW Receive Timestamp Extension Field is recommended to faithfully reflect the crystal rate, despite stepping or slewing a system clock. In case of link asymmetry, the Monotonic RAW Transmit Timestamp Extension Field is recommended in addition to the Monotonic RAW Receive Timestamp Extension Field. Measurements from the two extension fields can be used to identify link asymmetry and enhance time synchronization accuracy.

2. Conventions and Definitions

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.

2.1. Terminology

Monotonic Raw Original Timestamps (raw_org): Time of the monotonic raw clock at the client when the request departed for the server, in NTP timestamp format.

Monotonic Raw Receive Timestamps (raw_rec): Time of the monotonic raw clock at the server when the request arrived from the client, in NTP timestamp format.

Monotonic Raw Transmit Timestamps (raw_xmt): Time of the monotonic raw clock at the server when the response left for the client, in NTP timestamp format.

Monotonic Raw Destination Timestamps (raw_dst): Time of the monotonic raw clock at the client when the reply arrived from the server, in NTP timestamp format.

3. Monotonic Receive Timestamp Extension Field in NTPv5

The Monotonic Receive Timestamp Extension Field defined in NTPv5 uses a different clock to transfer frequency between client and server. In NTP version 4 (NTPv4) [RFC5905], the clock discipline function defines two methods to adjust system clock, i.e., step and slew. In the step mode, the clock is stepped to the correct offset. In the slew mode, the clock rate is adjusted to achieve the desired offset during a certain amount of time. The clock rate used to measure the Monotonic Receive Timestamp remains unchanged if the system clock is stepped, but is subject to changes if the clock is slewed, see Figure 1.

ClockID in Linux	STEP	SLEW
CLOCK_REALTIME	YES	YES
CLOCK_MONOTONIC	NO	YES
CLOCK_MONOTONIC_RAW	NO	NO

Figure 1 — Impact of NTP clock adjustment on clock rate/frequency.

In NTPv5 Use Cases and Requirements [I-D.ietf-ntp-ntpv5-requirements], it is recommended to adopt a linear and monotonic timescale when communicating time between a number of computers. Stepping a clock may cause the system time to jump backward, making the timescale non-monotonic. When the system clock is slewed, the rate of the monotonic clock source moves at the same speed as the system clock. The frequency-transfer offset can no longer reflect the rate of the crystal, thus, introducing errors in frequency transfer. In a multi-hop scenario, this effect can be amplified over a number of hops. In some scenarios, it can increase time errors when synchronizing time, sometimes, result in a system that fails to converge, see Section 5.

4. Monotonic RAW Timestamp Extension Fields

In the Linux system, CLOCK_MONOTONIC_RAW is a clock source that is not subject to NTP adjustment, despite stepping or slewing a clock, see Figure 1. It provides a stable source to calculate the frequency-transfer offset and reduces the error that has been introduced using the Monotonic Receive Timestamp extension field.

4.1. Monotonic Raw Receive Timestamp Extension Field

An NTPv5 message contains multiple optional extension fields. A Monotonic Raw Receive Timestamp Extension Field is recommended in addition to the Monotonic Receive Timestamp extension field. It is also recommended to derive the frequency-transfer offset from the Monotonic Raw Receive Timestamp Extension Field if CLOCK_MONOTONIC_RAW is available. The Monotonic Raw Receive Timestamp Extension Field has the same format of the Monotonic Receive Timestamp Extension Field. It complies to the constant length of 16 octets as defined in NTPv5. The counter and timestamp are set in response. This extension field enhances the accuracy of frequency-transfer function and further reduce synchronization time error.

4.2. Monotonic Raw Transmit Timestamp Extension Field

The Monotonic Raw Receive Timestamp Extension Field appears to be insufficient when dealing with asymmetric packet delay variation (PDV) on the forward and backward paths. The same issue exists with the Monotonic Receive Timestamp Extension Field. The Monotonic Raw Transmit Timestamp Extension Field is included to identify link asymmetry (i.e., different PDV on forward and backward paths) and reduce related errors. The Monotonic Raw Transmit Timestamp Extension Field has the same format of the Monotonic Receive Timestamp Extension Field.

5. Frequency to The Root Server Extension Field

In NTPv5, the frequency-transfer offset is computed as the offset of a client relative to its immediate preceding server. A client is able to synchronize with the primary server (i.e., the root server) only if its preceding server has synchronized its frequency with the primary server. The Frequency To The Root Server Extension Field is an optional field that can be used to expedite the convergence speed when synchronizing time. The Frequency To The Root Server Extension Field contains the frequency-transfer offset of a client relative to the Realtime clock of the primary server. This extension field has a fixed length of 12 octets. The 1-bit sign bit is a binary number indicates if the frequency of a client is faster (1) or slower (0) relative to the primary server. The absolute frequency-transfer offset relative to the primary server is carried by the remaining 31-bit.

```
  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 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |             Type              |            Length             |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |s|             Frequency To The Root Server (31)               |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2 Format of a Frequency To The Root Server Extension Field.

Assume a multi-hop scenario with n stratum levels. The primary server determines the frequency offset between the Realtime clock and the Monotonic Raw clock, and include this value into the Frequency To The Root Server Extension Field. The frequency-transfer offset of a server at stratum level i (2<i<=n) relative to its immediate preceding server is determined using the Monotonic Raw Receive Timestamp Extension Field (or with the Monotonic Receive Timestamp Extension Field if the clock is stepped). When receiving an NTP message, a server at stratum level i (2<i<=n) reads the Frequency To The Root Server Extension Field, and adds the frequency-transfer offset that it computed locally to the existing value. This way, the frequency-transfer offset of a server relative to the primary server is captured and passed down to the succeeding nodes.

For simplicity, assume that all nodes (other than the primary server) have successfully learnt their frequency-transfer offset relative to the preceding server at time t_k. Without the Frequency To The Root Server Extension Field, synchronization of a server to the primary server happens sequentially in time, i.e., a server with stratum level i needs to wait for its immediate preceding server (with stratum level i-1) to complete the operation of setting frequency-transfer offset. Note the operation of setting frequency offset is decoupled with NTP message exchange. This can take longer for the server at the last stratum level to synchronize with the primary server. With the assist of the Frequency To The Root Server Extension Field, the frequency-transfer offset of a server relative to the primary server can be immediately passed down in an NTP message, allowing the succeeding nodes to synchronize their frequency relative to the primary server once receiving that message.

6. Implementing the frequency transfer function with different timestamps

Frequency transfer function assisted with the Monotonic Receive Timestamp Extension Field NTPv5 does not disclose details on an algorithm that assist the implementation of the frequency transfer function in conjunction with the Monotonic Receive Timestamp Extension Field. One feasible implementation is to subtract the difference between two consecutive Monotonic Receive Timestamps from the difference between two consecutive Receive timestamps. This gives us the amount of correction that to be applied to the time-transfer offset. Once the time-transfer offset is corrected, the frequency-transfer offset can be derived using techniques such as linear regression from a number of time-transfer offset samples. When the system clock is stepped, the monotonic clock rate is unaffected. The above function will truthfully recover the frequency-transfer offset from a set of measurement samples. If the system clock is slewed, a client can note down the change of frequency over a certain amount of time. This allows the client to compensate the frequency change caused by clock slew. However, the Monotonic Receive Timestamp Extension Field is not able to recover the amount of slewed frequency in the remote server. This introduces errors while estimating the frequency-transfer offset. To the knowledge of the author, Chrony 4.7 release [Chrony-project] is the only open source implementation of the Monotonic Receive Timestamp extension field and the frequency transfer function defined in NTPv5. The performance of Chrony 4.7 in terms of frequency-transfer offset estimation would encounter the same issue if the system clock is slewed. In a multi-hop scenario, the frequency-transfer offset between a server and client pair may not converge. This will negatively impact the synchronization accuracy of following nodes.

6.1. Frequency transfer function assisted with the Monotonic Raw Receive Timestamp Extension Field

One approach to determine the frequency-transfer offset with the Monotonic Raw Timestamp Extension Field is at a client is to obtain the amount of offset correction that needs to be applied to the time-transfer offset. This can be done by subtracting the difference between two consecutively Monotonic Raw receive timestamps in the response from the difference between two consecutively Realtime timestamps. The frequency-transfer offset shall be estimated using techniques such as linear regression from a number of corrected time-transfer offset samples. Since the Monotonic Raw receive timestamps is not affected by NTP adjustment. No additional error is introduced if clock is slewed. Alternatively, a client can choose to determine the difference between the Monotonic Raw Receive Timestamps (T2_raw) in the extension field and the Monotonic Raw Original Timestamps

(T1_raw) maintained locally at the client. The frequency-transfer offset can be derived with a number of computed samples using techniques such as linear regression, etc.

6.2. Frequency transfer function assisted with the Monotonic Raw Transmit Timestamp Extension Field

The Monotonic Raw Transmit Timestamp Extension Field should always be used together with the Monotonic Raw Receive Timestamp Extension Field to enable bi-directional measurement. In case of link asymmetry, the forward and backward paths would experience different delay/jitter. Thus, T2_rwaw-T1_raw demonstrates distinct different behavior than T3_raw-T4_raw, i.e., difference between the Monotonic Raw Transmit Timestamps (T3_raw) in the extension field and the Monotonic Raw Destination Timestamps (T4_raw) at the client. The client can carefully apply a filter algorithm to take out anomalies on T2_raw-T1_raw and T3_raw-T4_raw respectively. Averaging over T’ 2_raw-T’ 1_raw and T’ 3_raw-T’ 4_raw gives us the offset and delay of the monotonic raw clock of a client relative to the server. the superscript ’ denotes data samples that are left after applying the filtering mechanism(s). The slope of the delay (or offset) derived from the monotonic raw timestamps would more truthfully reflect the true frequency-transfer offset between a server and client.

7. Security Considerations

As this document is intended to create discussion and consensus, it introduces no security considerations of its own.

8. IANA Considerations

This document has no IANA actions.

9. References

9.1. 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/rfc/rfc2119>.

[RFC5905] Mills, D. and Others, "Network Time Protocol Version 4: Protocol and Algorithms Specification", RFC 5905, DOI 10.17487/RFC5905, 2010, <https://www.rfc-editor.org/rfc/rfc5905>.

[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/rfc/rfc8174>.

9.2. Informative References

[Chrony-project] "Chrony 4.7", October 2025, <https://chrony-project.org>.

[I-D.draft-ietf-ntp-ntpv5] Lichvar, M. and T. Mizrahi, "Network Time Protocol Version 5", DOI Internet-Draft, draft-ietf-ntp-ntpv5-06, September 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-ntp-ntpv5-06>.

[I-D.ietf-ntp-ntpv5-requirements] Gruessing, J., "NTP Version 5 use cases and requirements", DOI Internet-Draft, draft-ietf-ntp-ntpv5-requirements-04, January 2024, <https://www.ietf.org/archive/id/draft-ietf-ntp-ntpv5-requirements-04.html>.

Authors' Addresses

Siyu Tang
Huawei Technologies
Email: siyutang@huawei.com

Guanhua Zhuang
Huawei Technologies
Email: zhuangguanhua@huawei.com