Workgroup: Network Time Protocol
Internet-Draft:
draft-ietf-ntp-ntpv5-requirements-03
Published: 16 September 2023
Intended Status: Informational
Expires: 19 March 2024
Authors: J. Gruessing
Nederlandse Publieke Omroep

NTPv5 Use Cases and Requirements

Abstract

This document describes the use cases, requirements, and considerations that should be factored in the design of a successor protocol to supersede version 4 of the NTP protocol [RFC5905] presently referred to as NTP version 5 ("NTPv5").

Note to Readers

RFC Editor: please remove this section before publication

Source code and issues for this draft can be found at https://github.com/fiestajetsam/draft-gruessing-ntp-ntpv5-requirements.

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 March 2024.

Copyright Notice

Table of Contents

1. Introduction

NTP version 4 [RFC5905] has seen active use for over a decade, and within this time period the protocol has not only been extended to support new requirements but has also fallen victim to vulnerabilities that have been used for distributed denial of service (DDoS) amplification attacks. In order to advance the protocol and address these known issues alongside add capabilities for future usage this document defines the current known and

applicable use cases in existing NTPv4 deployments and defines requirements for the future.

1.1. Notational Conventions

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.

Use of time specific terminology used in this document may further be specified in [RFC7384] or NTP specific terminology and concepts within [RFC5905].

2. Use Cases and Existing Deployments of NTP

There are several common scenarios for existing NTPv4 deployments: publicly accessible NTP services such as the NTP Pool [ntppool] are used to offer clock synchronisation for end users and embedded devices, ISP-provided servers are used to synchronise devices such as customer-premises equipment where reduced accuracy may be tolerable. Depending on the network and path these deployments may be affected by variable latency as well as throttling or blocking by providers.

Data centres and cloud computing providers have also deployed and offer NTP services both for internal use and for customers, particularly where the network is unable to offer or does not require other protocols e.g. PTP [IEEE-1588-2019], and where there may already be familiarity with NTP. As these deployments are less likely to be constrained by network latency or power the potential for higher levels of accuracy and precision within the bounds of the protocol are possible, particular through the use of modifications such as the use of bespoke algorithms.

3. Threat Analysis and Modeling

A considerable motivation towards a new version of the protocol is the inclusion of security primitives such as authentication and encryption to bring the protocol in-line with current best practices for protocol design.

There are numerous potential threats to a deployment or network handling traffic time synchronisation protocols that [RFC7384] section 3 describes, which can be summarised into three basic groups: Denial of Service (DoS), degradation of accuracy, and false time, all of which in various forms apply to NTP. However, not all threats apply specifically to NTP directly, most notable attacks on time sources (§3.2.10) and L2/L3 DoS Attacks (§3.2.7) as both are

outside the scope of the protocol, and the protocol itself cannot provide much in the way of mitigations.

3.1. Denial of Service and Amplification

NTPv4 has previously suffered from DDoS amplification attacks using a combination of IP address spoofing and private mode commands used in some NTP implementations, leading to an attacker being able to direct very large volumes of traffic to a victim IP address. Current mitigations are disabling private mode commands susceptible to attackes and encouraging network operators to implement BCP 38 [RFC2827] as well as source address validation where possible.

The NTPv5 protocol specification should be designed with current best practices for UDP based protocols in mind [RFC8085]. It should reduce the potential amplification factors in request/response payload sizes [drdos-amplification] through the use of padding of payload data, in addition to restricting command and diagnostic modes which could be exploited.

3.2. Accuracy Degradation

The risk that an on-path attacker can systemically delay packets between a client and server exists in all time protocols operating on insecure networks and its mitigations within the protocol are limited for a clock which is not yet synchronised. Increased path diversity and protocol support for synchronisation across multiple heterogeneous sources are likely the most effective mitigations.

3.3. False Time

Conversely, on-path attackers who can manipulate timestamps could also speed up a client's clock resulting in drift-related malfunctions and errors such as premature expiration of certificates on affected hosts. An attacker may also manipulate other data in flight to disrupt service and cause de-synchronisation. Additionally attacks via replaying transmitted packets can also delay or confuse receiving clocks impacting ongoing synchronisation.

Message authentication with regular key rotation should mitigate all of these cases; however deployments should consider finding an appropriate compromise between the frequency of rotation to balance the window of attack vs rate of re-keying.

4. Requirements

At a high level, NTPv5 should be a protocol that is capable of operating in local networks and over public internet connections where packet loss, delay, and filtering may occur. It should be able

to provide enough information for both basic time information and synchronisation.

4.1. Resource Management

Historically there have been many documented instances of NTP servers receiving ongoing large volumes of unauthorised traffic [ntp-misuse] and the design of NTPv5 must ensure the risk of these can be minimised through the use of signalling unwanted traffic (e.g Kiss of Death) or easily identifiable packet formats which make rate-limiting, filtering, or blocking by firewalls possible.

The protocol's loop avoidance mechanisms **SHOULD** be able to use identifiers that change over time and **MUST NOT** use identifiers tied to network topology. In particular such mechanism should not rely on any FQDN, IP address or identifier tied to a public certificate used or owned by the server. Servers **SHOULD** be able to migrate and change any identifier used as stratum topologies or network configuration changes occur.

An additional identifier mechanism **MAY** be considered for the purposes of client allow/deny lists, logging and monitoring. Such a mechanism when included, **SHOULD** be independent of any loop avoidance mechanism, and authenticity requirements **SHOULD** be considered.

The protocol **MUST** have the capability for servers to notify clients that the service is unavailable and clients **MUST** have clearly defined behaviours for honouring this signalling. In addition servers **SHOULD** be able to communicate to clients that they should reduce their query rate when the server is under high load or has reduced capacity.

Clients **SHOULD** periodically re-establish connections with servers to prevent maintaining prolonged connectivity to unavailable hosts and give operators the ability to move traffic away from hosts in a timely manner.

The protocol **SHOULD** have provisions for deployments where Network Address Translation occurs and define behaviours when NAT rebinding occurs. This should also not compromise any DDoS mitigation(s) that the protocol may define.

Client and server protocol modes **MUST** be supported, and other modes such as symmetric and broadcast **MAY** be supported by the protocol but **SHOULD NOT** be required by implementers to implement. Considerations should be made in these modes to avoid implementations and deployments from vulnerabilities and attacks.

4.2. Data Minimisation

To minimise ongoing use of deprecated fields and exposing identifying information of implementations and deployments, payload formats **SHOULD** use the least amount of fields and information where possible. The use of extensions should be preferred when transmitting optional data.

4.3. Algorithms

The use of algorithms describing functions such as clock filtering, selection, and clustering **SHOULD** have agility, allowing for implementations to develop and deploy new algorithms independently. Signalling of algorithm use or preference **SHOULD NOT** be transmitted by servers, however essential properties of the algorithm (e.g. precision) **SHOULD** be obvious.

The working group should consider creating a separate informational document to describe an algorithm to assist with implementation, and consider adopting future documents which describe new algorithms as they are developed. Specifying client algorithms separately from the protocol will allow NTPv5 to meet the needs of applications with a variety of network properties and performance requirements.

4.4. Timescales

The protocol should adopt a linear, monotonic timescale as the basis for communicating time. The format should provide sufficient scale, precision, and resolution to meet or exceed NTPv4's capabilities, and have a rollover date sufficiently far into the future that the protocol's complete obsolescence is likely to occur first. Ideally it should be similar or identical to the existing epoch and data model that NTPv4 defines to allow for implementations to better support both versions of the protocol, allowing for simpler implementations.

The timescale, in addition to any other time-sensitive information, **MUST** be sufficient to calculate representations of both UTC and TAI [TF.460-6], with UTC being the current timescale up to NTPv4. Through extensions the protocol **SHOULD** support additional timescale representations outside of the main specification, and all transmissions of time data **MUST** indicate the timescale in use.

4.5. Leap seconds

Transmission of UTC leap second information **MUST** be included in the protocol in order for clients to generate a UTC representation, but must be transmitted as separate information to the timescale. The specification **MUST** require that servers transmit upcoming leap seconds greater than 24 hours in linear timescale in advance if that

information is known by the server. If the server learns of a leap second less than 24 hours before an upcoming leap second event, it will start transmitting the information immediately.

Smearing [google-smear] of leap seconds **SHOULD** be supported in the protocol, and the protocol **MUST** support servers transmitting information if they are configured to smear leap seconds and if they are actively doing so. Behaviours for both client and server in handling leap seconds **MUST** be part of the specification; in particular how clients handle multiple servers where some may use leap seconds and others smearing, that servers should not apply both leap seconds and smearing, as well as details around smearing timescales. Supported smearing algorithms **MUST** be defined or referenced.

4.6. Backwards Compatibility with NTS and NTPv4

The desire for compatibility with older protocols should not prevent addressing deployment issues or cause ossification of the protocol caused by middleboxes [RFC9065].

Servers that support multiple versions of NTP **MUST** send a response in the same version as the request as the model of backwards compatibility. This does not preclude servers from acting as a client in one version of NTP and a server in another.

Protocol ossification **MUST** be addressed to prevent existing NTPv4 deployments which respond incorrectly to clients posing as NTPv5 from causing issues. Forward prevention of ossification (for a potential NTPv6 protocol in the future) should also be taken into consideration.

4.6.1. Dependent Specifications

Many other documents make use of NTP's data formats ([RFC5905] Section 6) for representing time, notably for media and packet timestamp measurements, such as SDP [RFC4566] and STAMP [RFC8762]. Any changes to the data formats should consider the potential implementation complexity that may be incurred.

4.7. Extensibility

The protocol **MUST** have the capability to be extended; implementations **MUST** ignore unknown extensions. Unknown extensions received from a lower stratum server **SHALL NOT** be re-transmitted towards higher stratum servers.

4.8. Security

Data authentication and integrity **MUST** be supported by the protocol, with optional support for data confidentiality. Downgrade attacks by an in-path attacker must be mitigated. The protocol **SHOULD** support different mechanisms to support different deployment use cases. Extensions and additional modes **SHOULD** also incorporate authentication and integrity on data which could be manipulated by an attacker, in-path or off-path.

Upgrading cryptographic algorithms must be supported, allowing for more secure cryptographic primitives to be incorporated as they are developed and as attacks and vulnerabilities with incumbent primitives are discovered.

Intermediate devices such as networking equipment capable of modifying NTP packets, for example to adjust timestamps **MUST** be able to do so without compromising authentication or confidentiality. Extension fields with separate authentication may be used to facilitate this.

Consideration must be given to how this will be incorporated into any applicable trust model. Downgrading attacks that could lead to an adversary disabling or removing encryption or authentication **MUST NOT** be possible in the design of the protocol.

5. Non-requirements

This section covers topics that are explicitly out of scope.

5.1. Server Malfeasance Detection

Detection and reporting of server malfeasance should remain out of scope as [I-D.ietf-ntp-roughtime] already provides this capability as a core functionality of the protocol.

5.2. Additional Time Information and Metadata

Previous versions of NTP do not transmit additional time information such as time zone data or historical leap seconds, and NTPv5 should not explicitly add support for it by default as existing protocols (e.g. TZDIST [RFC7808]) already provide mechanisms to do so. This does not prevent however, further extensions enabling this.

5.3. Remote Monitoring Support

Largely due to previous DDoS amplification attacks, mode 6 messages which have historically provided the ability for monitoring of servers **SHOULD NOT** be supported in the core of the protocol, however it may be provided as a separate extension specification. It is

likely that even with a new version of the protocol middleboxes may continue to block this mode in default configurations into the future.

6. IANA Considerations

This document makes no requests of IANA.

7. Security Considerations

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

8. References

8.1. Normative References

[I-D.ietf-ntp-roughtime] Malhotra, A., Langley, A., Ladd, W., and M. Dansarie, "Roughtime", Work in Progress, Internet-Draft, draft-ietf-ntp-roughtime-07, 26 September 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-ntp-roughtime-07>.

[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>.

[RFC2827] Ferguson, P. and D. Senie, "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", BCP 38, RFC 2827, DOI 10.17487/RFC2827, May 2000, <https://www.rfc-editor.org/rfc/rfc2827>.

[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/rfc/rfc5905>.

[RFC7384] Mizrahi, T., "Security Requirements of Time Protocols in Packet Switched Networks", RFC 7384, DOI 10.17487/

RFC7384, October 2014, <https://www.rfc-editor.org/rfc/rfc7384>.

[RFC8085] Eggert, L., Fairhurst, G., and G. Shepherd, "UDP Usage Guidelines", BCP 145, RFC 8085, DOI 10.17487/RFC8085, March 2017, <https://www.rfc-editor.org/rfc/rfc8085>.

[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>.

8.2. Informative References

[drdos-amplification] "Amplification and DRDoS Attack Defense -- A Survey and New Perspectives", n.d., <https://arxiv.org/abs/1505.07892>.

[google-smear] "Google Leap Smear", n.d., <https://developers.google.com/time/smear>.

[IEEE-1588-2019] "IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems", n.d..

[ntp-misuse] "NTP server misuse and abuse", n.d., <https://en.wikipedia.org/wiki/NTP_server_misuse_and_abuse>.

[ntppool] "pool.ntp.org: the internet cluster of ntp servers", n.d., <https://www.ntppool.org>.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, DOI 10.17487/RFC4566, July 2006, <https://www.rfc-editor.org/rfc/rfc4566>.

[RFC7808] Douglass, M. and C. Daboo, "Time Zone Data Distribution Service", RFC 7808, DOI 10.17487/RFC7808, March 2016, <https://www.rfc-editor.org/rfc/rfc7808>.

[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/rfc/rfc8762>.

[RFC9065] Fairhurst, G. and C. Perkins, "Considerations around Transport Header Confidentiality, Network Operations, and the Evolution of Internet Transport Protocols", RFC 9065,

DOI 10.17487/RFC9065, July 2021, <https://www.rfc-editor.org/rfc/rfc9065>.

[TF.460-6] "Standard-frequency and time-signal emissions", n.d., <https://www.itu.int/rec/R-REC-TF.460-6-200202-I/en>.

Appendix A. Acknowledgements

The author would like to thank Doug Arnold, Hal Murray, Paul Gear, and David Venhoek for contributions to this document, and would like to acknowledge Daniel Franke, Watson Ladd, Miroslav Lichvar for their existing documents and ideas. The author would also like to thank Angelo Moriondo, Franz Karl Achard, and Malcom McLean for providing the author with motivation.

Author's Address

James Gruessing
Nederlandse Publieke Omroep
Netherlands

Email: james.ietf@gmail.com