

Network Time Protocol (ntp) Working Group
Internet-Draft
Updates: [rfc5905](#) (if approved)
Intended status: Standards Track
Expires: April 24, 2020

F. Gont
G. Gont
SI6 Networks
M. Lichvar
Red Hat
October 22, 2019

Port Randomization in the Network Time Protocol Version 4
draft-ietf-ntp-port-randomization-00

Abstract

The Network Time Protocol can operate in several modes. Some of these modes are based on the receipt of unsolicited packets, and therefore require the use of a service/well-known port as the local port number. However, in the case of NTP modes where the use of a service/well-known port is not required, employing such well-known/service port unnecessarily increases the ability of attackers to perform blind/off-path attacks. This document formally updates [RFC5905](#), recommending the use of port randomization for those modes where use of the NTP service port is not required.

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 April 24, 2020.

Copyright Notice

Copyright (c) 2019 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

Internet-Draft

NTP Port Randomization

October 2019

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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	2
2.	Terminology	3
3.	Considerations About Port Randomization in NTP	3
3.1.	Mitigation Against Off-path Attacks	3
3.2.	Effects on Path Selection	4
3.3.	Filtering of NTP traffic	4
3.4.	Effect on NAT devices	5
3.5.	Relation to Other Mitigations for Off-Path Attacks	5
4.	Update to RFC5905	5
5.	Possible Future Work	6
6.	Implementation Status	6
7.	IANA Considerations	7
8.	Security Considerations	7
9.	Acknowledgments	8
10.	References	8
10.1.	Normative References	8
10.2.	Informative References	8
	Authors' Addresses	10

[1.](#) Introduction

The Network Time Protocol (NTP) is one of the oldest Internet protocols, and currently specified in [[RFC5905](#)]. Since its original implementation, standardization, and deployment, a number of vulnerabilities have been found both in the NTP specification and in some of its implementations [[NTP-VULN](#)]. Some of these vulnerabilities allow for off-path/blind attacks, where an attacker can send forged packets to one or both NTP peers for achieving Denial of Service (DoS), time-shifts, and other undesirable outcomes. Many of these attacks require the attacker to guess or know at least a target NTP association, typically identified by the tuple {srcaddr, srcport, dstaddr, dstport, keyid}. Some of these parameters may be easily known or guessed.

NTP can operate in several modes. Some of these modes rely on the ability of nodes to receive unsolicited packets, and therefore require the use of a service/well-known port number. However, for modes where the use of a service/well-known port is not required, employing such well-known/service port improves the ability of an

attacker to perform blind/off-path attacks (since knowledge of such port number is typically required for such attacks). A recent study [[NIST-NTP](#)] that analyzes the port numbers employed by NTP clients suggests that a considerable number of NTP clients employ the NTP service/well-known port as their local port, or select predictable ephemeral port numbers, thus improving the ability of attackers to perform blind/off-path attacks against NTP.

[BCP 156](#) [[RFC6056](#)] already recommends the randomization of transport-protocol ephemeral ports. This document aligns NTP with the recommendation in [BCP 156](#) [[RFC6056](#)], by formally updating [[RFC5905](#)] such that port randomization is employed for those NTP modes for which the use of the NTP service port is not required.

[2.](#) Terminology

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 [[RFC2119](#)].

[3.](#) Considerations About Port Randomization in NTP

The following subsections analyze a number of considerations about transport-protocol port randomization when applied to NTP.

[3.1.](#) Mitigation Against Off-path Attacks

There has been a fair share of work in the area of off-path/blind attacks against transport protocols and upper-layer protocols, such as [[RFC5927](#)] and [[RFC4953](#)]. Whether the target of the attack is a transport protocol instance (e.g., TCP connection) or an upper-layer protocol instance (e.g., an application protocol instance), the attacker is required to know or guess the five-tuple {Protocol, IP Source Address, IP Destination Address, Source Port, Destination Port} that identifies the target transport protocol instance or the transport protocol instance employed by the target upper-layer

protocol instance. Therefore, increasing the difficulty of guessing this five-tuple helps mitigate blind/off-path attacks.

As a result of this considerations, [BCP 156](#) [[RFC6056](#)] recommends the randomization of transport-protocol ephemeral ports. And as such, this document aims to bring the NTP specification [[RFC5905](#)] in line with the aforementioned recommendation.

We note that the use of port randomization is a transport-layer mitigation against off-path/blind attacks, and does not preclude (nor is it precluded by), other possible mitigations for off-path attacks that might be implemented by an application protocol (e.g.

[[I-D.ietf-ntp-data-minimization](#)]). For instance, some of the aforementioned mitigations may be ineffective against some off-path attacks [[NTP-FRAG](#)] or may benefit from the additional entropy provided by port randomization [[NTP-security](#)].

[3.2.](#) Effects on Path Selection

Intermediate systems implementing the Equal-Cost Multi-Path (ECMP) algorithm may select the outgoing link by computing a hash over a number of values, that include the transport-protocol source port. Thus, as discussed in [[NTP-CHLNG](#)], the selected client port may have an influence on the measured delay and jitter values.

This might mean, for example, that two systems in the same network that synchronize their clocks with the same NTP server might end up with a significant offset between their clocks as a result of their NTP samples taking paths with very different characteristics.

If port randomization is applied for every NTP request, requests/responses would be distributed over the different available paths, including those with the smallest delay. The clock filter algorithm could readily select one of such samples with lowest delays, in the same way that the clock selection and clock cluster algorithms might also end up selecting other time sources with smaller resulting dispersion. On the other hand, if port-randomization is applied on a per-association basis, in scenarios where the aforementioned ECMP algorithm is employed, request/responses to the same association would likely follow the same path, since the IP addresses and transport port numbers employed for an association would not change.

[Section 4](#) recommends NTP implementations to randomize the ephemeral port number of non-symmetrical associations on a per-association basis (as opposed to "per-transaction"), since this more conservative approach avoids the possible negative implications of port randomization on time synchronization.

[3.3.](#) Filtering of NTP traffic

In a number of scenarios (such as when mitigating DDoS attacks), a network operator may want to differentiate between NTP requests sent by clients, and NTP responses sent by NTP servers. If an implementation employs the NTP service port for the client port number, requests/responses cannot be readily differentiated by inspecting the source and destination port numbers. Implementation of port randomization for non-symmetrical modes allows for simple differentiation of NTP requests and responses, and for the enforcement of security policies that may be valuable for the mitigation of DDoS attacks.

Gont, et al.

Expires April 24, 2020

[Page 4]

Internet-Draft

NTP Port Randomization

October 2019

[3.4.](#) Effect on NAT devices

Some NAT devices will not translate the source port of a packet when a privileged port number is employed. In networks where such NAT devices are employed, use of the NTP service port for the client port will essentially limit the number of hosts that may successfully employ NTP client implementations.

In the case of NAT devices that will translate the source port even when a privileged port is employed, packets reaching the external realm of the NAT will not employ the NTP service port as the local port, since the local port will normally be translated by the NAT device possibly, but not necessarily, with a random port.

[3.5.](#) Relation to Other Mitigations for Off-Path Attacks

Ephemeral Port Randomization is a best current practice ([BCP 156](#)) that helps mitigate off-path attacks at the transport-layer. It is orthogonal to other possible mitigations for off-path attacks that may be implemented at other layers (such as the use of timestamps in NTP) which may or may not be effective against some off-path attacks (see e.g. [\[NTP-FRAG\]](#)). This document aligns NTP with the existing

best current practice on ephemeral port selection, irrespective of other techniques that may (and should) be implemented for mitigating off-path attacks.

4. Update to [RFC5905](#)

The following text from [Section 9.1](#) ("Peer Process Variables") of [\[RFC5905\]](#):

dstport: UDP port number of the client, ordinarily the NTP port number PORT (123) assigned by the IANA. This becomes the source port number in packets sent from this association.

is replaced with:

dstport: UDP port number of the client. In the case of broadcast server mode (5) and symmetric modes (1 and 2), it must contain the NTP port number PORT (123) assigned by the IANA. In other cases, it SHOULD contain a randomized port number, as specified in [\[RFC6056\]](#). The value in this variable becomes the source port number of packets sent from this association.

NOTES:

When port randomization is employed, the port number must be randomized on a per-association basis. That is, a random port number is selected when an association is first mobilized, and the selected port number is expected to remain constant during the life of an association.

On most current operating systems (that implement ephemeral port randomization [\[RFC6056\]](#)), an NTP client may normally rely on the operating system for performing port randomization. For example, NTP implementations employing the Sockets API may achieve port randomization by *not* specifying the local port for the corresponding socket, or `bind()`ing the local socket to the "special" port 0 (which for the Sockets API has the special meaning of "any port"). `connect()`ing the docket will make the port inaccessible by other systems (that is, only packets from

the specified remote socket will be received by the application).

5. Possible Future Work

Port numbers could be randomized on a per-association basis, or on a per-request basis. When the port number is randomized on a per-association basis, a random port number is selected when an association is first mobilized, and the selected port remains constant during the life of the association. On the other hand, when the port number is randomized on a per-request basis, each client request will (statistically) employ a different ephemeral port for each request. As discussed in [Section 3](#), varying the port number across requests may impact the time quality achieved with NTP. As a result, this document recommends the conservative approach of randomizing port numbers on a per-association basis (as opposed to a "per-transaction" basis). The possibility of randomizing port numbers on a per-transaction may be subject of future work, and is not recommended by this document.

6. Implementation Status

[RFC Editor: Please remove this section before publication of this document as an RFC.]

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in [\[RFC7942\]](#). The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was

supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

OpenNTPD:

[\[OpenNTPD\]](#) has never explicitly set the local port of NTP clients, and thus employs the ephemeral port selection algorithm

implemented by the operating system. Thus, on all operating systems that implement port randomization (such as current versions of OpenBSD, Linux, and FreeBSD), OpenNTPD will employ port randomization for client ports.

chrony:

[[chrony](#)] has never explicitly set the local port of NTP clients, and thus employs the ephemeral port selection algorithm implemented by the operating system. Thus, on all operating systems that implement port randomization (such as current versions of OpenBSD, Linux, and FreeBSD), chrony will employ port randomization for client ports.

nwtime.org's sntp client:

sntp does not explicitly set the local port, and thus employs the ephemeral port selection algorithm implemented by the operating system. Thus, on all operating systems that implement port randomization (such as current versions of OpenBSD, Linux, and FreeBSD), it will employ port randomization for client ports.

7. IANA Considerations

There are no IANA registries within this document. The RFC-Editor can remove this section before publication of this document as an RFC.

8. Security Considerations

The security implications of predictable numeric identifiers [[I-D.gont-predictable-numeric-ids](#)] (and of predictable transport-protocol port numbers [[RFC6056](#)] in particular) have been known for a long time now. However, the NTP specification has traditionally followed a pattern of employing common settings and code even when not strictly necessary, which at times has resulted in negative security and privacy implications (see e.g. [[I-D.ietf-ntp-data-minimization](#)]). The use of the NTP service port (123) for the srcport and dstport variables is not required for all operating modes, and such unnecessary usage comes at the expense of reducing the amount of work required for an attacker to successfully perform off-path/blind attacks against NTP. Therefore, this document

protocol port randomization when use of the NTP service port is not required.

This issue has been tracked by US-CERT with VU#597821, and has been assigned CVE-2019-11331.

9. Acknowledgments

Watson Ladd raised the problem of DDoS mitigation when the NTP service port is employed as the client port (discussed in [Section 3.3](#) of this document).

The authors would like to thank (in alphabetical order) Ivan Arce, Todd Glassey, Watson Ladd, Aanchal Malhotra, Danny Mayer, Gary E. Miller, Dieter Sibold, Steven Sommars, and Ulrich Windl, for providing valuable comments on earlier versions of this document.

The authors would like to thank Harlan Stenn for answering questions about nwttime.org's NTP implementation.

Fernando would like to thank Nelida Garcia and Jorge Oscar Gont, for their love and support.

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>>.
- [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>>.
- [RFC6056] Larsen, M. and F. Gont, "Recommendations for Transport-Protocol Port Randomization", [BCP 156](#), [RFC 6056](#), DOI 10.17487/RFC6056, January 2011, <<https://www.rfc-editor.org/info/rfc6056>>.

10.2. Informative References

- [chrony] "chrony", <<https://chrony.tuxfamily.org/>>.

[I-D.gont-predictable-numeric-ids]

Gont, F. and I. Arce, "Security and Privacy Implications of Numeric Identifiers Employed in Network Protocols", [draft-gont-predictable-numeric-ids-03](#) (work in progress), March 2019.

[I-D.ietf-ntp-data-minimization]

Franke, D. and A. Malhotra, "NTP Client Data Minimization", [draft-ietf-ntp-data-minimization-04](#) (work in progress), March 2019.

[NIST-NTP]

Sherman, J. and J. Levine, "Usage Analysis of the NIST Internet Time Service", Journal of Research of the National Institute of Standards and Technology Volume 121, March 2016, <<https://tf.nist.gov/general/pdf/2818.pdf>>.

[NTP-CHLNG]

Sommars, S., "Challenges in Time Transfer Using the Network Time Protocol (NTP)", Proceedings of the 48th Annual Precise Time and Time Interval Systems and Applications Meeting, Monterey, California pp. 271-290, January 2017, <http://leapsecond.com/ntp/NTP_Paper_Sommars_PTTI2017.pdf>.

[NTP-FRAG]

Malhotra, A., Cohen, I., Brakke, E., and S. Goldberg, "Attacking the Network Time Protocol", NDSS'17, San Diego, CA. Feb 2017, 2017, <<http://www.cs.bu.edu/~goldbe/papers/NTPattack.pdf>>.

[NTP-security]

Malhotra, A., Van Gundy, M., Varia, V., Kennedy, H., Gardner, J., and S. Goldberg, "The Security of NTP's Datagram Protocol", Cryptology ePrint Archive Report 2016/1006, 2016, <<https://eprint.iacr.org/2016/1006>>.

[NTP-VULN]

Network Time Foundation, "Security Notice", Network Time Foundation's NTP Support Wiki , <<https://support.ntp.org/bin/view/Main/SecurityNotice>>.

[OpenNTPD]

"OpenNTPD Project", <<https://www.openntpd.org>>.

[RFC4953] Touch, J., "Defending TCP Against Spoofing Attacks",

[RFC 4953](#), DOI 10.17487/RFC4953, July 2007,
<<https://www.rfc-editor.org/info/rfc4953>>.

Gont, et al.

Expires April 24, 2020

[Page 9]

Internet-Draft

NTP Port Randomization

October 2019

[RFC5927] Gont, F., "ICMP Attacks against TCP", [RFC 5927](#),
DOI 10.17487/RFC5927, July 2010,
<<https://www.rfc-editor.org/info/rfc5927>>.

[RFC7942] Sheffer, Y. and A. Farrel, "Improving Awareness of Running
Code: The Implementation Status Section", [BCP 205](#),
[RFC 7942](#), DOI 10.17487/RFC7942, July 2016,
<<https://www.rfc-editor.org/info/rfc7942>>.

Authors' Addresses

Fernando Gont
SI6 Networks
Evaristo Carriego 2644
Haedo, Provincia de Buenos Aires 1706
Argentina

Phone: +54 11 4650 8472
Email: fgont@si6networks.com
URI: <https://www.si6networks.com>

Guillermo Gont
SI6 Networks
Evaristo Carriego 2644
Haedo, Provincia de Buenos Aires 1706
Argentina

Phone: +54 11 4650 8472
Email: ggont@si6networks.com
URI: <https://www.si6networks.com>

Miroslav Lichvar
Red Hat
Purkynova 115
Brno 612 00
Czech Republic

Email: mlichvar@redhat.com

Gont, et al.

Expires April 24, 2020

[Page 10]