Network Working Group K. Fujiwara
Internet-Draft JPRS
Intended status: Best Current Practice P. Vixie
Expires: 26 August 2021 Farsight
22 February 2021

Fragmentation Avoidance in DNS
draft-ietf-dnsop-avoid-fragmentation-04

Abstract

EDNS0 enables a DNS server to send large responses using UDP and is widely deployed. Path MTU discovery remains widely undeployed due to security issues, and IP fragmentation has exposed weaknesses in application protocols. Currently, DNS is known to be the largest user of IP fragmentation. It is possible to avoid IP fragmentation in DNS by limiting response size where possible, and signaling the need to upgrade from UDP to TCP transport where necessary. This document proposes to avoid IP fragmentation in DNS.

Table of Contents

1. Introduction

DNS has EDNS0 [RFC6891] mechanism. It enables a DNS server to send large responses using UDP. EDNS0 is now widely deployed, and DNS (over UDP) is said to be the biggest user of IP fragmentation.

However, "Fragmentation Considered Poisonous" [Herzberg2013] proposed effective off-path DNS cache poisoning attack vectors using IP fragmentation. "IP fragmentation attack on DNS" [Hlavacek2013] and "Domain Validation++ For MitM-Resilient PKI" [Brandt2018] proposed that off-path attackers can intervene in path MTU discovery [RFC1191] to perform intentionally fragmented responses from authoritative servers. [RFC7739] stated the security implications of predictable fragment identification values.

DNSSEC is a countermeasure against cache poisoning attacks that use IP fragmentation. However, DNS delegation responses are not signed with DNSSEC, and DNSSEC does not have a mechanism to get the correct response if an incorrect delegation is injected. This is a denial-of-service vulnerability that can yield failed name resolutions. If cache poisoning attacks can be avoided, DNSSEC validation failures will be avoided.

In Section 3.2 (Message Side Guidelines) of UDP Usage Guidelines [RFC8085] we are told that an application SHOULD NOT send UDP datagrams that result in IP packets that exceed the Maximum Transmission Unit (MTU) along the path to the destination.

A DNS message receiver cannot trust fragmented UDP datagrams primarily due to the small amount of entropy provided by UDP port numbers and DNS message identifiers, each of which being only 16 bits in size, and both likely being in the first fragment of a packet, if fragmentation occurs. By comparison, TCP protocol stack controls packet size and avoid IP fragmentation under ICMP NEEDFRAG attacks. In TCP, fragmentation should be avoided for performance reasons, whereas for UDP, fragmentation should be avoided for resiliency and authenticity reasons.

[RFC8900] summarized that IP fragmentation introduces fragility to Internet communication. The transport of DNS messages over UDP should take account of the observations stated in that document.

TCP avoids fragmentation using its Maximum Segment Size (MSS) parameter, but each transmitted segment is header-size aware such that the size of the IP and TCP headers is known, as well as the far end's MSS parameter and the interface or path MTU, so that the segment size can be chosen so as to keep the each IP datagram below a target size. This takes advantage of the elasticity of TCP's packetizing process as to how much queued data will fit into the next segment. In contrast, DNS over UDP has little datagram size elasticity and lacks insight into IP header and option size, and so must make more conservative estimates about available UDP payload space.

This document proposes to set IP_DONTFRAG / IPV6_DONTFRAG in DNS/UDP messages in order to avoid IP fragmentation, and describes how to avoid packet losses due to IP_DONTFRAG / IPV6_DONTFRAG.

2. Terminology

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 BCP14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

"Requestor" refers to the side that sends a request. "Responder" refers to an authoritative, recursive resolver or other DNS component that responds to questions. (Quoted from EDNS0 [RFC6891])

"Path MTU" is the minimum link MTU of all the links in a path between a source node and a destination node. (Quoted from [RFC8201])

"Path MTU discovery" is defined by [RFC1191], [RFC8201] and [RFC8899].

Many of the specialized terms used in this document are defined in DNS Terminology [RFC8499].

3. Proposal to avoid IP fragmentation in DNS

The methods to avoid IP fragmentation in DNS are described below:

3.1. Recommendations for UDP responders

* UDP responders SHOULD send DNS responses with IP_DONTFRAG / IPV6_DONTFRAG [RFC3542] options.

* If the UDP responder detects immediate error that the UDP packet cannot be sent beyond the path MTU size (EMSGSIZE), the UDP responder MAY recreate response packets fit in path MTU size, or TC bit set.

* UDP responders MAY probe to discover the real MTU value per destination.

* UDP responders SHOULD compose UDP responses that result in IP packets that do not exceed the path MTU to the requestor. If the path MTU discovery failed or is impossible, UDP responders SHOULD compose UDP responses that result in IP packets that do not exceed the default maximum DNS/UDP payload size described in Section 3.3.

  The cause and effect of the TC bit is unchanged from EDNS0 [RFC6891].

3.2. Recommendations for UDP requestors

* UDP requestors SHOULD send DNS requests with IP_DONTFRAG / IPV6_DONTFRAG [RFC3542] options.

* UDP requestors MAY probe to discover the real MTU value per destination. Then, calculate their maximum DNS/UDP payload size as the reported path MTU minus IPv4/IPv6 header size (20 or 40) minus UDP header size (8). If the path MTU discovery failed or is impossible, use the default maximum DNS/UDP payload size described in Section 3.3.

* UDP requestors SHOULD use the requestor's payload size as the calculated or the default maximum DNS/UDP payload size.

* UDP requestors MAY drop fragmented DNS/UDP responses without IP reassembly to avoid cache poisoning attacks.

* DNS responses may be dropped by IP fragmentation. Upon a timeout, UDP requestors may retry using TCP or UDP, per local policy.

3.3. Default Maximum DNS/UDP payload size

Default maximum DNS/UDP payload size for IPv6 is XXXX. (Choose 1232, 1400, 1472 or other good values before/at WGLC)

Default maximum DNS/UDP payload size for IPv4 is XXXX. (Choose 1232, 1400, 1452 or other good values before/at WGLC)

Operators of DNS servers SHOULD measure their path MTU to well-known locations on the Internet, such as [a-m].root-servers.net or [a-m].gtld-servers.net at setting up the servers. The smallest value of path MTU is the server's path MTU to the Internet. The server's maximum DNS/UDP payload size for IPv4 is the reported path MTU minus IPv4 header size (20) minus UDP header size (8). The server's maximum DNS/UDP payload size for IPv6 is the reported path MTU minus IPv6 header size (40) minus UDP header size (8).

Discussions under here will be moved to appendix as a background of default maximum DNS/UDP payload size when the discussion is over.

There are many discussions for default path MTU size and maximum DNS/UDP payload size.

* The minimum MTU for an IPv6 interface is 1280 octets (see Section 5 of [RFC8200]). Then, we can use it as default path MTU value for IPv6.

* Most of the Internet and especially the inner core has an MTU of at least 1500 octets. An operator of a full resolver would be well advised to measure their path MTU to several authority name servers and to a random sample of their expected stub resolver client networks, to find the upper boundary on IP/UDP packet size in the average case. This limit should not be exceeded by most messages received or transmitted by a full resolver, or else fallback to TCP will occur too often. An operator of authoritative servers would also be well advised to measure their path MTU to several full-service resolvers. The Linux tool "tracepath" can be used to measure the path MTU to well known authority name servers such as [a-m].root-servers.net or [a-m].gtld-servers.net. If the reported path MTU is for example no smaller than 1460, then the maximum DNS/UDP payload would be 1432 for IP4 (which is 1460 - IP4 header(20) - UDP header(8)) and 1412 for IP6 (which is 1460 - IP6 header(40) - UDP header(8)). To allow for possible IP options and distant tunnel overhead, a useful default for maximum DNS/UDP payload size would be 1400.

* [RFC4035] defines that "A security-aware name server MUST support the EDNS0 message size extension, MUST support a message size of at least 1220 octets". Then, the smallest number of the maximum DNS/UDP payload size is 1220.

* In order to avoid IP fragmentation, [DNSFlagDay2020] proposed that the UDP requestors set the requestor's payload size to 1232, and the UDP responders compose UDP responses fit in 1232 octets. The size 1232 is based on an MTU of 1280, which is required by the IPv6 specification [RFC8200], minus 48 octets for the IPv6 and UDP headers.

* [Huston2021] analyzed the result of [DNSFlagDay2020], reported that their measurements suggest that in the interior of the Internet between recursive resolvers and authoritative servers the prevailing MTU is at 1,500 and there is no measurable signal of use of smaller MTUs in this part of the Internet, and proposed that their measurements suggest setting the EDNS0 Buffer size to IPv4 1472 octets and IPv6 1452 octets.

4. Incremental deployment

The proposed method supports incremental deployment.

When a full-service resolver implements the proposed method, its stub resolvers (clients) and the authority server network will no longer observe IP fragmentation or reassembly from that server, and will fall back to TCP when necessary.

When an authoritative server implements the proposed method, its full service resolvers (clients) will no longer observe IP fragmentation or reassembly from that server, and will fall back to TCP when necessary.

5. Request to zone operators and DNS server operators

Large DNS responses are the result of zone configuration. Zone operators SHOULD seek configurations resulting in small responses. For example,

* Use smaller number of name servers (13 may be too large)

* Use smaller number of A/AAAA RRs for a domain name

* Use 'minimal-responses' configuration: Some implementations have 'minimal responses' configuration that causes DNS servers to make response packets smaller, containing only mandatory and required data (Appendix B).

* Use smaller signature / public key size algorithm for DNSSEC. Notably, the signature size of ECDSA or EdDSA is smaller than RSA.

6. Considerations

6.1. Protocol compliance

In prior research ([Fujiwara2018] and dns-operations mailing list discussions), there are some authoritative servers that ignore EDNS0 requestor's UDP payload size, and return large UDP responses.

It is also well known that there are some authoritative servers that do not support TCP transport.

Such non-compliant behavior cannot become implementation or configuration constraints for the rest of the DNS. If failure is the result, then that failure must be localized to the non-compliant servers.

7. IANA Considerations

This document has no IANA actions.

8. Security Considerations

9. Acknowledgments

The author would like to specifically thank Paul Wouters, Mukund Sivaraman Tony Finch and Hugo Salgado for extensive review and comments.

10. References

10.1. Normative References

[RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191, DOI 10.17487/RFC1191, November 1990, <https://www.rfc-editor.org/info/rfc1191>.

[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>.

[RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Protocol Modifications for the DNS Security Extensions", RFC 4035, DOI 10.17487/RFC4035, March 2005, <https://www.rfc-editor.org/info/rfc4035>.

[RFC5155] Laurie, B., Sisson, G., Arends, R., and D. Blacka, "DNS Security (DNSSEC) Hashed Authenticated Denial of Existence", RFC 5155, DOI 10.17487/RFC5155, March 2008, <https://www.rfc-editor.org/info/rfc5155>.

[RFC6891] Damas, J., Graff, M., and P. Vixie, "Extension Mechanisms for DNS (EDNS(0))", STD 75, RFC 6891, DOI 10.17487/RFC6891, April 2013, <https://www.rfc-editor.org/info/rfc6891>.

[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/info/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/info/rfc8174>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

[RFC8201] McCann, J., Deering, S., Mogul, J., and R. Hinden, Ed., "Path MTU Discovery for IP version 6", STD 87, RFC 8201, DOI 10.17487/RFC8201, July 2017, <https://www.rfc-editor.org/info/rfc8201>.

[RFC8499] Hoffman, P., Sullivan, A., and K. Fujiwara, "DNS Terminology", BCP 219, RFC 8499, DOI 10.17487/RFC8499, January 2019, <https://www.rfc-editor.org/info/rfc8499>.

[RFC8899] Fairhurst, G., Jones, T., Tüxen, M., Rüngeler, I., and T. Völker, "Packetization Layer Path MTU Discovery for Datagram Transports", RFC 8899, DOI 10.17487/RFC8899, September 2020, <https://www.rfc-editor.org/info/rfc8899>.

[RFC8900] Bonica, R., Baker, F., Huston, G., Hinden, R., Troan, O., and F. Gont, "IP Fragmentation Considered Fragile", BCP 230, RFC 8900, DOI 10.17487/RFC8900, September 2020, <https://www.rfc-editor.org/info/rfc8900>.

10.2. Informative References

[Brandt2018] Brandt, M., Dai, T., Klein, A., Shulman, H., and M. Waidner, "Domain Validation++ For MitM-Resilient PKI", Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security , 2018.

[DNSFlagDay2020] "DNS flag day 2020", n.d., <https://dnsflagday.net/2020/>.

[Fujiwara2018] Fujiwara, K., "Measures against cache poisoning attacks using IP fragmentation in DNS", OARC 30 Workshop , 2019.

[Herzberg2013] Herzberg, A. and H. Shulman, "Fragmentation Considered Poisonous", IEEE Conference on Communications and Network Security , 2013.

[Hlavacek2013] Hlavacek, T., "IP fragmentation attack on DNS", RIPE 67 Meeting , 2013, <https://ripe67.ripe.net/presentations/240-ipfragattack.pdf>.

[Huston2021] Huston, G. and J. Damas, "Measuring DNS Flag Day 2020", OARC 34 Workshop , February 2021.

[RFC3542] Stevens, W., Thomas, M., Nordmark, E., and T. Jinmei, "Advanced Sockets Application Program Interface (API) for IPv6", RFC 3542, DOI 10.17487/RFC3542, May 2003, <https://www.rfc-editor.org/info/rfc3542>.

[RFC7739] Gont, F., "Security Implications of Predictable Fragment Identification Values", RFC 7739, DOI 10.17487/RFC7739, February 2016, <https://www.rfc-editor.org/info/rfc7739>.

Appendix A. How to retrieve path MTU value to a destination from applications

Socket options: "IP_MTU (since Linux 2.2) Retrieve the current known path MTU of the current socket. Valid only when the socket has been connected. Returns an integer. Only valid as a getsockopt(2)." (Quoted from Debian GNU Linux manual: ip(7))

"IPV6_MTU getsockopt(): Retrieve the current known path MTU of the current socket. Only valid when the socket has been connected. Returns an integer." (Quoted from Debian GNU Linux manual: ipv6(7))

Appendix B. Minimal-responses

Some implementations have 'minimal responses' configuration that causes a DNS server to make response packets smaller, containing only mandatory and required data.

Under the minimal-responses configuration, DNS servers compose response messages using only RRSets corresponding to queries. In case of delegation, DNS servers compose response packets with delegation NS RRSet in authority section and in-domain (in-zone and below-zone) glue in the additional data section. In case of non-existent domain name or non-existent type, the start of authority (SOA RR) will be placed in the Authority Section.

In addition, if the zone is DNSSEC signed and a query has the DNSSEC OK bit, signatures are added in answer section, or the corresponding DS RRSet and signatures are added in authority section. Details are defined in [RFC4035] and [RFC5155].

Authors' Addresses

Kazunori Fujiwara
Japan Registry Services Co., Ltd.
Chiyoda First Bldg. East 13F, 3-8-1 Nishi-Kanda, Chiyoda-ku, Tokyo
101-0065
Japan

Phone: +81 3 5215 8451
Email: fujiwara@jprs.co.jp

Paul Vixie
Farsight Security Inc
177 Bovet Road, Suite 180
San Mateo, CA, 94402
United States of America

Phone: +1 650 393 3994
Email: vixie@fsi.io