Network Working Group
Internet-Draft
Intended status: Experimental
Expires: January 20, 2019
A. Mayrhofer
nic.at GmbH
July 19, 2018

Padding Policy for EDNS(0)
draft-ietf-dprive-padding-policy-06

Abstract

RFC 7830 specifies the EDNS(0) 'Padding' option, but does not specify the actual padding length for specific applications. This memo lists the possible options ("Padding Policies"), discusses implications of each of these options, and provides a recommended (experimental) option.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

[RFC7830] specifies the Extensions Mechanisms for DNS (EDNS(0)) "Padding" option, which allows DNS clients and servers to artificially increase the size of a DNS message by a variable number of bytes, hampering size-based correlation of encrypted DNS messages.

However, RFC 7830 deliberately does not specify the actual length of padding to be used. This memo discusses options regarding the actual size of padding, lists advantages and disadvantages of each of these "Padding Strategies", and provides a recommended (experimental) strategy.

Padding DNS messages is useful only when transport is encrypted, using protocols such as DNS over Transport Layer Security [RFC7858], DNS over Datagram Transport Layer Security [RFC8094] or other encrypted DNS transports specified in the future.

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. General Guidance

EDNS(0) options space: The maximum message length as dictated by the protocol limits the space for EDNS(0) options. Since padding will reduce the message space available to other EDNS(0) options, "Padding" MUST be the last EDNS(0) option applied before a DNS message is sent.

Resource Conservation: Especially in situations where networking and processing resources are scarce (e.g. battery powered long-life devices, low bandwidth or high cost links), the tradeoff between increased size of padded DNS messages and the corresponding gain in confidentiality must be carefully considered.

Transport Protocol Independence: The message size used as input to the various padding strategies MUST be calculated excluding the potential extra 2-octet length field used in TCP transport. Otherwise, the padded (observable) size of the DNS packets could significantly change between different transport protocols, and reveal an indication of the original (unpadded) length. For example, given a "Block Length" padding strategy with a block length of 32 octets, and a DNS message with a size of 59 octets, the message would be padded to 64 octets when transported over UDP. If that same message was transported over TCP, and the padding strategy would consider the extra 2 octets of the length field (61 octets in total), the padded message would be 96 octets long (as the minimum length of the Padding option is 4 octets).

4. Padding Strategies

This section contains a recommended strategy, as well as a non-exhaustive list of other sensible strategies in choosing padding length. Note that, for completeness, Appendix A contains two more (non-sensible) strategies.

4.1. Block Length Padding - Recommended Strategy

Based on empirical research performed by Daniel K. Gillmor [dkg-padding-ndss], EDNS Padding SHOULD be performed following the "Block Length Padding" strategy as follows:

(1) Clients SHOULD pad queries to the closest multiple of 128 octets.

(2) If a Server receives a query that includes the EDNS(0) Padding Option, it MUST pad the corresponding response (See Section 4 of RFC7830) and SHOULD pad the corresponding response to a multiple of 468 octets (see below).

Note that the recommendation above applies only if the DNS transport is encrypted (See Section 6 of RFC 7830).

In Block Length Padding, a sender pads each message so that its padded length is a multiple of a chosen block length. This creates a greatly reduced variety of message lengths. An implementor needs to consider that even the zero-length EDNS(0) Padding Option increases the length of the packet by 4 octets.

Options: Block Length - for queries, values between 16 and 128 octets were discussed before empiric research was performed. Responses will require larger block sizes (see [dkg-padding-ndss] and above for a discussion).

Very large block lengths will have confidentiality properties similar to the "Maximal Length Padding" strategy (Section 4.2.1), since almost all messages will fit into a single block. Such "very large block length" values are 288 bytes for the query (the maximum size of a one-question query over TCP, without any EDNS(0) options), and the EDNS(0) buffer size of the server for the responses.

Advantages: This policy is reasonably easy to implement, reduces the variety of message ("fingerprint") sizes significantly, and does not require a source of (pseudo) random numbers, since the padding length required can be derived from the actual (unpadded) message.

Disadvantage: Given an unpadded message and the block size of the padding (which is assumed to be public knowledge once a server is reachable), the size range of a padded message can be predicted. Therefore, the minimum length of the unpadded message can be infered.

The empirical research cited above performed a simulation of padding, based on real-world DNS traffic captured on busy recursive resolvers of a research network. The evaluation of the performance of individual padding policies was based on a "cost to attacker" and "cost to defender" function, where the "cost to attacker" was defined as the percentage of query/response pairs falling into the same size bucket, and "cost to defender" as the size factor between padded and unpadded messages. Padding with a block size of 128 bytes on the query side, and 468 bytes on the response side was considered the

optimum trade-off between defender and attacker cost. The response block size of 468 was chosen so that 3 blocks of 468 octets would still comfortably fit into typical Maximum Transmission Unit (MTU) size values.

The Block Size will interact with the MTU size. Especially for length values that are a large fraction of the MTU, unless the block length is chosen so that a multiple just fits into the MTU, Block Padding may cause unneccessary fragmentation for UDP based delivery. Also, chosing a block length larger than the MTU of course always forces to always fragment.

Note: Once DNSSEC validating clients become more prevalent, observed size patterns are expected to change significantly. In such case, the recommended strategy might need to be revisited.

4.2. Other Strategies

4.2.1. Maximal Length Padding

In Maximal Length Padding the sender pads every message to the maximum size as allowed by protocol negotiations.

Advantages: Maximal Length Padding, when combined with encrypted transport, provides the highest possible level of message size confidentiality.

Disadvantages: Maximal Length Padding is wasteful, and requires resources on the client, all intervening network and equipment, and the server. Depending on the negotiated size, this strategy will commonly exceed the MTU, and then result in a consistent number of fragments reducing delivery probability when datagram based transport (such as UDP) is used.

Due to resource consumption, Maximal Length Padding is NOT RECOMMENDED.

4.2.2. Random Length Padding

When using Random Length Padding, a sender pads each message with a random amount of padding. Due to the size of the EDNS(0) Padding Option itself, each message size is hence increased by at least 4 octets. The upper limit for padding is the maximum message size. However, a client or server may choose to impose a lower maximum padding length.

Options: Maximum and minimum padding length.

Advantages: Theoretically, this policy should create a natural "distribution" of message sizes.

Disadvantage: Random Length padding allows an attacker who can observe a large number of requests to infer the length of the original value by observing the distribution of total lengths.

According to the limited empirical data available, Random Length Padding exposes slightly more entropy to an attacker than Block Length Padding. Due to that, and the risk outlined above, Random Length Padding is NOT RECOMMENDED.

4.2.3. Random Block Length Padding

This policy combines Block Length Padding with a random component. Specifically, a sender randomly chooses between a few block length values and then applies Block Length Padding based on the chosen block length. The random selection of block length might even be reasonably based on a "weak" source of randomness, such as the transaction ID of the message.

Options: Number of and the values for the set of Block Lengths, source of "randomness"

Advantages: Compared to Block Length Padding, this creates more variety in the resulting message sizes for a certain individual original message length.

Disadvantage: Requires more implementation effort compared to simple Block Length Padding

Random Block Length Padding (as other combinations of padding strategies) requires further empirical study.

5. Acknowledgements

Daniel K. Gillmor performed empirical research out of which the "Recommended Strategy" was copied. Stephane Bortzmeyer and Hugo Connery provided text. Shane Kerr, Sara Dickinson, Paul Hoffman, Magnus Westerlund, Charlie Kaufman, Joe Clarke and Meral Shirazipour performed reviews or provided substantial comments.

6. IANA Considerations

This document has no considerations for IANA.

7. Security Considerations

The choice of the right padding policy (and the right parameters for the chosen policy) has a significant impact on the resilience of encrypted DNS against size-based correlation attacks. Therefore, any implementor of EDNS(0) Padding must carefully consider which policies to implement, the default policy chosen, which parameters to make configurable, and the default parameter values.

No matter how carefully a client selects their Padding policy, this effort can be jeopardized if the server chooses to apply an ineffective Padding policy to the corresponding response packets. Therefore, a client applying Padding may want to choose a DNS server which does apply at least an equally effective Padding policy on responses.

Note that even with encryption and padding, it might be trivial to identify that the observed traffic is DNS. Also, padding does not prevent information leak via other side channels (particularly timing information and number of query/response pairs). Counter-measures against such other side channels could include injecting artificial "cover traffic" into the stream of DNS messages, or delaying DNS responses by a certain amount of jitter. Such strategies are out of scope of this document. Additionally, there is neither enough theoretic analysis nor experimental data available to recommend any such countermeasures.

8. Changes

[Note to RFC Editors: This whole section is to be removed before publication]

8.1. draft-ietf-dprive-padding-policy-06

Changes based on IESG evaluation: Removed duplicate paragraph about MTU impact, switched Terminology boilerplate to RFC8174, changed text regarding Random Padding, changed text regarding very large block paddings, some minor edits.

8.2. draft-ietf-dprive-padding-policy-05

Changes based on outcomes of IETF-wide LC + various reviews: Meral Shirazipour (Gen-ART), Charlie Kaufmann (SECDIR), Joe Clarke (OPSDIR - changed document flow based on comments),

8.3. draft-ietf-dprive-padding-policy-04

Changes based on WGLC: Changed implementor consideration text in Security Con section (Sara), moved "No Padding" and "Fixed Length Padding" to appendix (Stephane, Paul), Changed TODO in Random Padding to info from empirical study (Stephen), Added note to pad only if transport encrypted (Stephen), added intro text referencing to DNSoTLS and DNSoDTLS (Stephane), added text about timing/jitter to security considerations.

8.4. draft-ietf-dprive-padding-policy-03

Editorial changes in various spots. Added text about excluding TCP length field, more security considerations, addressing Sara's other feedback to -02.

8.5. draft-ietf-dprive-padding-policy-02

Changed Document Status to Experimental, added "maximum length" padding policy, reworded "block length" policy, some editorial changes.

8.6. draft-ietf-dprive-padding-policy-01

Some (mostly editorial) changes to text. Added "Recommendation" section based on dkg's research.

8.7. draft-ietf-dprive-padding-policy-00

Initial (mostly unmodified) WG version. Changed "Profile" to "Policy" to avoid confusion with the (D)TLS profiles document.

8.8. draft-mayrhofer-dprive-padding-profiles-00

Initial version

9. References

9.1. Normative References

[dkg-padding-ndss]
Gillmor, D., "Empirical DNS Padding Policy", March 2017, <https://dns.cmrg.net/ndss2017-dprive-empirical-DNS-traffic-size.pdf>.

[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>.

[RFC7830] Mayrhofer, A., "The EDNS(0) Padding Option", RFC 7830, DOI 10.17487/RFC7830, May 2016, <https://www.rfc-editor.org/info/rfc7830>.

[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>.

9.2. Informative References

[RFC7858] Hu, Z., Zhu, L., Heidemann, J., Mankin, A., Wessels, D., and P. Hoffman, "Specification for DNS over Transport Layer Security (TLS)", RFC 7858, DOI 10.17487/RFC7858, May 2016, <https://www.rfc-editor.org/info/rfc7858>.

[RFC8094] Reddy, T., Wing, D., and P. Patil, "DNS over Datagram Transport Layer Security (DTLS)", RFC 8094, DOI 10.17487/RFC8094, February 2017, <https://www.rfc-editor.org/info/rfc8094>.

Appendix A. Non-sensible Padding Policies

A.1. No Padding

In the "No Padding" policy, the EDNS0 Padding option is not used, and the size of the final (actually, "non-padded") message obviously exactly matches the size of the unpadded message. Even though this "non-policy" seems redundant in this list, its properties must be considered for cases where just one of the parties (client or server) applies padding.

Also, this "policy" is required when the remaining message size of the unpadded message does not allow for the Padding option to be included (less than 4 octets left).

Advantages: This "policy" requires no additional resources on client, server and network side.

Disadvantages: The original size of the message remains unchanged, hence this approach provides no additional confidentiality.

"No Padding" MUST NOT be used unless message size disallows the use of Padding.

A.2. Fixed Length Padding

In fixed length padding, a sender chooses to pad each message with a padding of constant length.

Options: Actual length of padding

Advantages: Since the padding is constant in length, this policy is very easy to implement, and at least ensures that the message length diverges from the length of the original packet (even only by a fixed value)

Disadvantage: Obviously, the amount of padding easily discoverable from a single unencrypted message, or by observing message patterns. When a public DNS server applies this policy, the length of the padding hence must be assumed to be public knowledge. Therefore, this policy is (almost) as useless as the "No Padding" option described above.

"Fixed Length Padding" MUST NOT be used except for test applications.

Author's Address

Alexander Mayrhofer
nic.at GmbH
Karlsplatz 1/2/9
Vienna 1010
Austria

Email: alex.mayrhofer.ietf@gmail.com
URI: http://edns0-padding.org/