Network Working Group A. Mayrhofer
Internet-Draft nic.at GmbH
Intended status: Experimental September 27, 2017
Expires: March 31, 2018

Padding Policy for EDNS(0)
draft-ietf-dprive-padding-policy-02

Abstract

RFC 7830 specifies the EDNS0 'Padding' option, but does not specify the actual padding length for specific applications. This memo lists the possible options ("Padding Policies"), discusses the implications of each of these options, and provides a recommended (experimental) option.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

RFC 7830 [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.

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 [RFC2119].

3. General Guidance

Padding DNS messages does not have any semantic impact on the DNS protocol. However, the length of (possible) padding does depend on the circumstances under which a DNS message is created, specifically the maximum message length as dictated by protocol negotiations. Since padding may frustrate the message space available to other EDNS options, "Padding" MUST be the last EDNS option applied before a DNS message is sent.

Especially in situations with scarce computing and networking resources such as long-life battery powered devices, the tradeoff between significantly increasing the size of DNS messages by generous padding and the corresponding gain in confidentiality must be carefully considered.

4. Padding Strategies

This section is a non-exhaustive list of possible strategies in choosing padding length

4.1. 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.

4.2. 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 experimental applications.

4.3. Block Length Padding

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 EDNS0 Padding Option increases the length of the packet by 4 octets.

Options: Block Length - values between 16 and 128 octets for the queries seem reasonable, responses will require larger block sizes (see [dkg-padding-ndss] and Section 5 for a discussion).

Very large block lengths will have confidentiality properties similar to the "Maximum Length Padding" strategy (Section 4.4), since almost all messages will fit into a single block. In that case, reasonable values may be 288 bytes for the query (the maximum size of a one-question query over TCP, without any EDNS0 options), and the EDNS 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 of a padded message can be predicted. Therefore, minimum and maximum length of the unpadded message are known.

Block Length Padding is the currently RECOMMENDED strategy (see Section 5).

4.4. Maximal Lenth Padding ('The Full Monty')

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.

Maximal Length Padding is NOT RECOMMENDED.

4.5. 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 EDNS0 Padding Option itself, each message size is hence increased by at least 4 octets. The upper limit for pading is the maximum message size. However, a client or server may choose to impose a lower maximum padding length.

Options: Maximum (and eventually minimum) padding length.

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

Disadvantage: This policy requires a good source of (pseudo) keeping up with the required message rates. Especially on busy servers, this may be a significant hindrance.

TODO: Recommendation - this is (at first glance) the best policy, but requires significant effort

4.6. Random Block Length Padding

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

Options: Number of size of 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. Also, compared to "Random Length Padding", it might not require a "full blown" random number source.

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

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

5. Recommended Strategy

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

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

(2) If a Server sees padding in a query, it SHOULD pad the corresponding response to a multiple of 468 octects.

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 MTU values.

6. 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 performed reviews and provided substantial comments.

7. IANA Considerations

This document has no considerations for IANA.

8. Security Considerations

The choice of the right padding policy (and the right parameters for the chose policy) has a significant impact on the resilience of encrypted DNS against size-based correlation attacks. Therefore, any implementor of EDNS0 Padding must carefully consider the chosen policy and its parameters.

A clients carefully chosen Padding policy may be without effect if the corresponding server does apply an inffective (or no) Padding policy on the response packets. Therefore, a client applying Padding may want to chose a DNS server which does apply at least an equally effective Padding policy on responses.

9. Changes

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

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

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

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

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

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

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

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

Initial version

10. 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>.

Author's Address

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

Email: alex.mayrhofer.ietf@gmail.com