Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 3, 2019

W. Kumari
Google
E. Hunt
ISC
R. Arends
ICANN
W. Hardaker
USC/ISI
D. Lawrence
Akamai Technologies

July 02, 2018

Extended DNS Errors
draft-ietf-dnsop-extended-error-01

Abstract

This document defines an extensible method to return additional information about the cause of DNS errors. The primary use case is to extend SERVFAIL to provide additional information about the cause of DNS and DNSSEC failures.

[Open question: The document currently defines a registry for errors. It has also been suggested that the option also carry human readable (text) messages, to allow the server admin to provide additional debugging information (e.g: "example.com pointed their NS at us. No idea why...", "We don't provide recursive DNS to 192.0.2.0. Please stop asking...", "Have you tried Acme Anvil and DNS? We do DNS right..." (!). Please let us know if you think text is needed, or if a 16bit FCFS registry is expressive enough.]

[Open question: This document discusses extended *errors*, but it has been suggested that this could be used to also annotate *non-error* messages. The authors do not think that this is a good idea, but could be persuaded otherwise.]

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/.

Copyright Notice

Table of Contents

1. Introduction and background

There are many reasons that a DNS query may fail, some of them transient, some permanent; some can be resolved by querying another server, some are likely best handled by stopping resolution. Unfortunately, the error signals that a DNS server can return are very limited, and are not very expressive. This means that applications and resolvers often have to "guess" at what the issue is - e.g the answer was marked REFUSED because of a lame delegation, or because of a lame delegation or because the nameserver is still starting up and loading zones? Is a SERVFAIL a DNSSEC validation issue, or is the nameserver experiencing a bad hair day?

A good example of issues that would benefit by additional error information is an error caused by a DNSSEC validation issue. When a stub resolver queries a DNSSEC bogus name (using a validating resolver), the stub resolver receives only a SERVFAIL in response. Unfortunately, SERVFAIL is used to signal many sorts of DNS errors, and so the stub resolver simply asks the next configured DNS resolver. The result of trying the next resolver is one of two outcomes: either the next resolver also validates, a SERVFAIL is returned again, and the user gets an (largely) incomprehensible error message; or the next resolver is not a validating resolver, and the user is returned a potentially harmful result.

This document specifies a mechanism to extend (or annotate) DNS errors to provide additional information about the cause of the error. This information can be used by the resolver to make a decision regarding whether or not to retry, or by technical users attempting to debug issues.

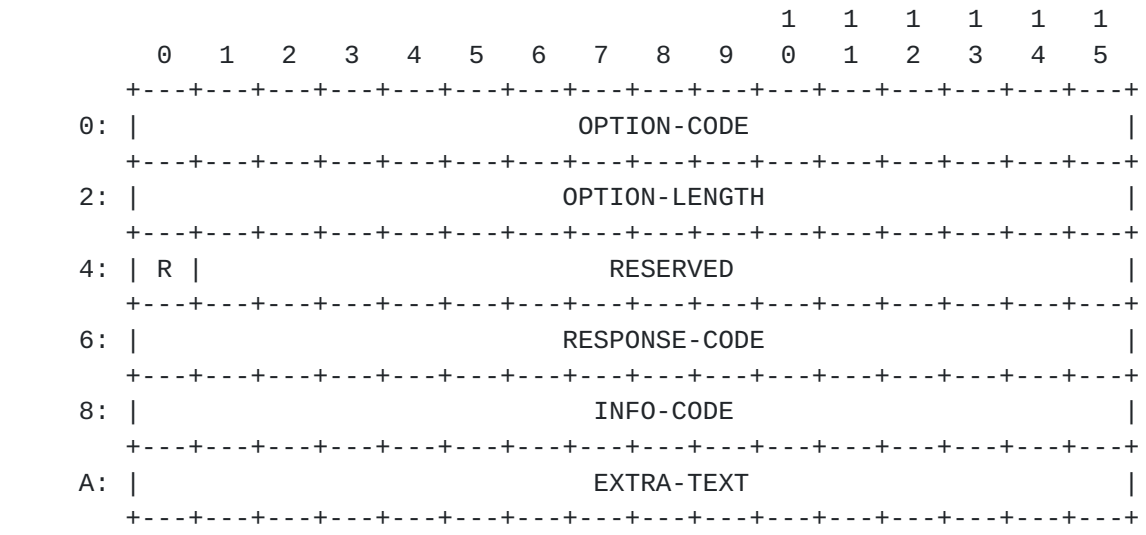
Here is a reference to an "external" (non-RFC / draft) thing: ([IANA.AS_Numbers]). And this is a link to an ID:[I-D.ietf-sidr-iana-objects].

1.1. Requirements notation

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].

2. Extended Error EDNS0 option format

This draft uses an EDNS0 ([RFC2671]) option to include extended error (ExtError) information in DNS messages. The option is structured as follows:



- OPTION-CODE, 2 octets (defined in [RFC6891]), for ExtError is TBD.

- OPTION-LENGTH, 2 octets ((defined in [RFC6891]) contains the length of the payload (everything after OPTION-LENGTH) in octets and should be 4.

- RESERVED, 2 octets; the first bit (R) indicates a flag defined in this specification. The remaining bits are reserved for future use, potentially as additional flags.

- RESPONSE-CODE, 2 octets: this SHOULD be a copy of the RCODE from the primary DNS packet. When including multiple extended error EDNS0 records in a response in order to provide additional error information, the RESPONSE-CODE MAY be a different RCODE.

- INFO-CODE, 2 octets.

- A variable length EXTRA-TEXT field holding additional textual information. It may be zero length when no additional textual information is included.

Currently the only defined flag is the R flag.

R - Retry The R (or Retry) flag provides a hint to the receiver that it should retry the query, probably by querying another server. If the R bit is set (1), the sender believes that retrying the

query may provide a successful answer next time; if the R bit is clear (0), the sender believes that it should not ask another server.

The remaining bits in the RESERVED field are reserved for future use and MUST be set to 0 by the sender and SHOULD be ignored by the receiver.

INFO-CODE: A code point that, when combined with the RCODE from the DNS packet, serve as a joint-index into the IANA "Extended DNS Errors" registry.

3. Use of the Extended DNS Error option

The Extended DNS Error (EDE) is an EDNS option. It can be included in any error response (SERVFAIL, NXDOMAIN, REFUSED, etc) to a query that includes an EDNS option. This document includes a set of initial codepoints (and requests to the IANA to add them to the registry), but is extensible via the IANA registry to allow additional error and information codes to be defined in the future.

The R (Retry) flag provides a hint (or suggestion) as to what the receiver may want to do with this annotated error. The mechanism is specifically designed to be extensible, and so implementations may receive EDE codes that it does not understand. The R flag allows implementations to make a decision as to what to do if it receives a response with an unknown code - retry or drop the query. Note that this flag is only a suggestion or hint. Receivers can choose to ignore this hint.

The EXTRA-INFO textual field may be zero-length, or may hold additional information useful to network operators.

4. Defined Extended DNS Errors

This document defines some initial EDE codes. The mechanism is intended to be extensible, and additional codepoints will be registered in the "Extended DNS Errors" registry. This document provides suggestions for the R flag, but the originating server may ignore these recommendations if it knows better.

The RESPONSE-CODE and the INFO-CODE from the EDE EDNS option is used to serve as a double index into the "Extended DNS Error codes" IANA registry, the initial values for which are defined in the following sub-sections.

4.1. SERVFAIL(3) extended information codes

4.1.1. Extended DNS Error Code 1 - DNSSEC Bogus

The resolver attempted to perform DNSSEC validation, but validation ended in the Bogus state. The R flag should not be set.

4.1.2. Extended DNS Error Code 2 - DNSSEC Indeterminate

The resolver attempted to perform DNSSEC validation, but validation ended in the Indeterminate state. The R flag should not be set.

4.1.3. Extended DNS Error Code 3 - Signature Expired

The resolver attempted to perform DNSSEC validation, but the signature was expired. The R flag should not be set.

4.1.4. Extended DNS Error Code 4 - Signature Not Yet Valid

The resolver attempted to perform DNSSEC validation, but the signatures received were not yet valid. The R flag should not be set.

4.1.5. Extended DNS Error Code 5 - Unsupported DNSKEY Algorithm

The resolver attempted to perform DNSSEC validation, but a DNSKEY RRSET contained only unknown algorithms. The R flag should not be set.

4.1.6. Extended DNS Error Code 6 - Unsupported DS Algorithm

The resolver attempted to perform DNSSEC validation, but a DS RRSET contained only unknown algorithms. The R flag should not be set.

4.1.7. Extended DNS Error Code 7 - DNSKEY missing

A DS record existed at a parent, but no DNSKEY record could be found for the child. The R flag should not be set.

4.1.8. Extended DNS Error Code 8 - RRSIGs missing

The resolver attempted to perform DNSSEC validation, but no RRSIGs could be found for at least one RRset where RRSIGs were expected.

4.1.9. Extended DNS Error Code 9 - No Zone Key Bit Set

The resolver attempted to perform DNSSEC validation, but no Zone Key Bit was set in a DNSKEY.

4.2. REFUSED(5) extended information codes

4.2.1. Extended DNS Error Code 1 - Lame

An authoritative resolver that receives a query (with the RD bit clear) for a domain for which it is not authoritative SHOULD include this EDE code in the REFUSED response. Implementations should set the R flag in this case (another nameserver might not be lame).

4.2.2. Extended DNS Error Code 2 - Prohibited

An authoritative or recursive resolver that receives a query from an "unauthorized" client can annotate its REFUSED message with this code. Examples of "unauthorized" clients are recursive queries from IP addresses outside the network, blacklisted IP addresses, local policy, etc.

Implementations SHOULD allow operators to define what to set the R flag to in this case.

5. IANA Considerations

[This section under construction, beware.]

5.1. new Extended Error Code EDNS Option

This document defines a new EDNS(0) option, entitled "Extended DNS Error", assigned a value of TBD1 from the "DNS EDNS0 Option Codes (OPT)" registry [to be removed upon publication: [http://www.iana.org/assignments/dns-parameters/dns-parameters.xhtml#dns-parameters-11]

Value	Name	Status	Reference
TBD	Extended DNS Error	TBD	[This document]

5.2. new Extended Error Code EDNS Option

This document defines a new double-index IANA registry table, where the first index value is the RCODE value and the second index value is the INFO-CODE from the Extended DNS Error EDNS option defined in this document. The IANA is requested to create and maintain this

"Extended DNS Error codes" registry. The codepoint space for each RCODE index is to be broken into 3 ranges:

- 1 - 16384: Specification required.
- 16385 - 65000: First Come First Served
- 65000 - 65534: Experimental / Private use

The codepoints 0, 65535 are reserved.

A starting table, based on the contents of this document, is as follows:

RCODE	EDE-INFO-CODE	Meaning	Ref
SERVFAIL(2)	DNSSEC_BOGUS(1)	DNSSEC Validation resulted in Bogus	section <xref target="errbogus" />
SERVFAIL(2)	DNSSEC_INDETERMINATE(2)	DNSSEC Validation resulted in Indeterminate	section <xref target="errindeterminate" />

[incomplete]

6. Open questions

1. Can this be included in *any* response or only responses to requests that included an EDNS option? Resolvers are supposed to ignore additional. EDNS capable ones are supposed to simply ignore unknown options. I know the spec says you can only include EDNS0 in a response if in a request -- it is time to reevaluate this?

2. Can this be applied to *any* response, or only error responses?

3. Should textual information be allowed as well? What if the only thing allowed is a domain name, e.g to point at where validation began failing?

7. Security Considerations

DNSSEC is being deployed - unfortunately a significant number of clients (~11% according to [GeoffValidation]), when receiving a

SERVFAIL from a validating resolver because of a DNSSEC validaion issue simply ask the next (non-validating) resolver in their list, and don't get any of the protections which DNSSEC should provide. This is very similar to a kid asking his mother if he can have another cookie. When the mother says "No, it will ruin your

dinner!", going off and asking his (more permissive) father and getting a "Yes, sure, cookie!".

8. Acknowledgements

The authors wish to thank Geoff Huston and Bob Harold, Carlos M. Martinez, Peter DeVries, George Michelson, Mark Andrews, Ondrej Sury, Edward Lewis, Paul Vixie, Shane Kerr. They also vaguely remember discussing this with a number of people over the years, but have forgotten who all they were -- if you were one of them, and are not listed, please let us know and we'll acknowledge you.

I also want to thank the band "Infected Mushroom" for providing a good background soundtrack (and to see if I can get away with this!) Another author would like to thank the band "Mushroom Infectors". This was funny at the time we wrote it, but I cannot remember why...

We would like to especially thank Peter van Dijk, who sent GitHub pull requests.

9. References

9.1. Normative References

[IANA.AS_Numbers]
IANA, "Autonomous System (AS) Numbers", <http://www.iana.org/assignments/as-numbers>.

[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>.

9.2. Informative References

[GeoffValidation]
IANA, "A quick review of DNSSEC Validation in today's Internet", June 2016, <http://www.potaroo.net/presentations/2016-06-27-dnssec.pdf>.

[I-D.ietf-sidr-iana-objects]
Manderson, T., Vegoda, L., and S. Kent, "RPKI Objects issued by IANA", draft-ietf-sidr-iana-objects-03 (work in progress), May 2011.

Appendix A. Changes / Author Notes.

[RFC Editor: Please remove this section before publication]

From -00 to -01:

- Address comments from IETF meeting.
- document copying the response code
- mention zero length fields are ok
- clarify lookup procedure
- mention that table isn't done

From -03 to -IETF 00:

- Renamed to draft-ietf-dnsop-extended-error

From -02 to -03:

- Added David Lawrence -- I somehow missed that in last version.

From -00 to -01;

- Fixed up some of the text, minor clarifications.

Authors' Addresses

Warren Kumari
Google
1600 Amphitheatre Parkway
Mountain View, CA 94043
US

Email: warren@kumari.net

Evan Hunt
ISC
950 Charter St
Redwood City, CA 94063
US

Email: each@isc.org

Roy Arends
ICANN

Email: roy.arends@icann.org

Wes Hardaker
USC/ISI
P.O. Box 382
Davis, VA 95617
US

Email: ietf@hardakers.net

David C Lawrence
Akamai Technologies
150 Broadway
Cambridge, MA 02142-1054
US

Email: tale@akamai.com