Internet Engineering Task Force (IETF)
Request for Comments: 6975
Category: Standards Track
ISSN: 2070-1721

S. Crocker
Shinkuro Inc.
S. Rose
NIST

July 2013

Signaling Cryptographic Algorithm Understanding in DNS Security Extensions (DNSSEC)

Abstract

The DNS Security Extensions (DNSSEC) were developed to provide origin authentication and integrity protection for DNS data by using digital signatures. These digital signatures can be generated using different algorithms. This document specifies a way for validating end-system resolvers to signal to a server which digital signature and hash algorithms they support. The extensions allow the signaling of new algorithm uptake in client code to allow zone administrators to know when it is possible to complete an algorithm rollover in a DNSSEC-signed zone.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 5741.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at http://www.rfc-editor.org/info/rfc6975.

Table of Contents

1. Introduction

The DNS Security Extensions (DNSSEC), [RFC4033], [RFC4034], and [RFC4035], were developed to provide origin authentication and integrity protection for DNS data by using digital signatures. Each digital signature (RRSIG) Resource Record (RR) contains an algorithm code number that corresponds to a DNSSEC public key (DNSKEY) RR. These algorithm codes tell validators which cryptographic algorithm was used to generate the digital signature.

Likewise, the Delegation Signer (DS) RRs and Hashed Authenticated Denial of Existence (NSEC3) RRs use a hashed value as part of their resource record data (RDATA) and, like digital signature algorithms, these hash algorithms have code numbers. All three algorithm codes (RRSIG/DNSKEY, DS, and NSEC3) are maintained in unique IANA registries.

This document sets specifies a way for validating end-system resolvers to tell a server in a DNS query which digital signature and/or hash algorithms they support. This is done using the new Extension Mechanisms for DNS (EDNS0) options specified below in Section 2 for use in the OPT meta-RR [RFC6891]. These three new EDNS0 option codes are all OPTIONAL to implement and use.

These proposed EDNS0 options serve to measure the acceptance and use of new digital signing algorithms. These signaling options can be used by zone administrators as a gauge to measure the successful deployment of code that implements the newly deployed digital signature algorithm, DS hash, and the NSEC3 hash algorithm used with DNSSEC. A zone administrator is able to determine when to stop signing with a superseded algorithm when the server sees that a significant number of its clients signal that they are able to accept the new algorithm. Note that this survey may be conducted over a period of years before a tipping point is seen.

This document does not seek to introduce another process for including new algorithms for use with DNSSEC. It also does not address the question of which algorithms are to be included in any official list of mandatory or recommended cryptographic algorithms for use with DNSSEC. Rather, this document specifies a means by which a client query can signal the set of algorithms and hashes that it implements.

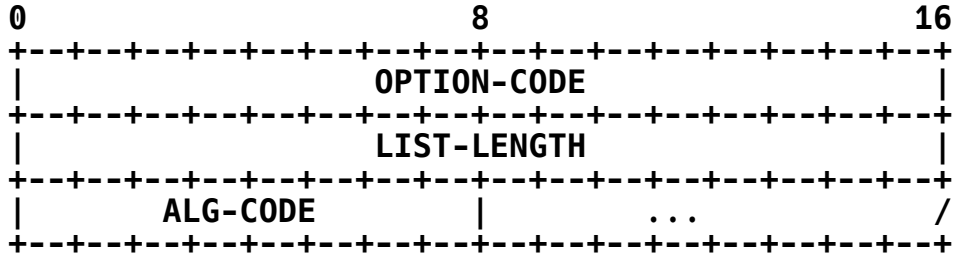
2. Requirements Language

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 RFC 2119 [RFC2119].

3. Signaling DNSSEC Algorithm Understood (DAU), DS Hash Understood (DHU), and NSEC3 Hash Understood (N3U) Using EDNS

The EDNS0 specification outlined in [RFC6891] defines a way to include new options using a standardized mechanism. These options are contained in the RDATA of the OPT meta-RR. This document defines three new EDNS0 options for a client to signal which digital signature and/or hash algorithms the client supports. These options can be used independently of each other and MAY appear in any order in the OPT RR. Each option code can appear only once in an OPT RR.

The figure below shows how each option is defined in the RDATA of the OPT RR specified in [RFC6891]:



OPTION-CODE is the code for the given signaling option. The options are:

- DNSSEC Algorithm Understood (DAU) option for DNSSEC digital signing algorithms. Its value is fixed at 5.
- DS Hash Understood (DHU) option for DS RR hash algorithms. Its value is fixed at 6.
- NSEC3 Hash Understood (N3U) option for NSEC3 hash algorithms. Its value is fixed at 7.

LIST-LENGTH is the length of the list of digital signatures or hash algorithm codes in octets. Each algorithm code occupies a single octet.

ALG-CODE is the list of assigned values of DNSSEC zone signing algorithms, DS hash algorithms, or NSEC3 hash algorithms (depending on the OPTION-CODE in use) that the client declares to be supported. The order of the code values can be arbitrary and MUST NOT be used to infer preference.

If all three options are included in the OPT RR, there is a potential for the OPT RR to take up considerable size in the DNS message. However, in practical terms, including all three options is likely to take up 22-32 octets (average of 6-10 digital signature algorithms, 3-5 DS hash algorithms, and 1-5 NSEC3 hash algorithms) including the EDNS0 option codes and option lengths in potential future examples.

4. Client Considerations

A validating end-system resolver sets the DAU, DHU, and/or N3U option, or combination thereof, in the OPT meta-RR when sending a query. The validating end-system resolver MUST also set the DNSSEC OK bit [RFC4035] to indicate that it wishes to receive DNSSEC RRs in the response.

Note that the PRIVATEDNS (253) and/or the PRIVATEOID (254) digital signature codes both cover a potentially wide range of algorithms and are likely not useful to a server. There is no compelling reason for a client to include these codes in its list of the DAU. Likewise, clients MUST NOT include RESERVED codes in any of the options. Additionally, a client is under no obligation to list every algorithm it implements and MAY choose to only list algorithms the client wishes to signal as understood.

Since the DAU, DHU, and/or N3U options are only set in the query, if a client sees these options in the response, no action needs to be taken and the client MUST ignore the option values.

4.1. Stub Resolvers

Typically, stub resolvers rely on an upstream recursive server (or cache) to provide a response. So optimal setting of the DAU, DSU, and N3U options depends on whether the stub resolver elects to perform its own validation.

4.1.1. Validating Stub Resolvers

A validating stub resolver sets the DNSSEC OK (DO) bit [RFC4035] to indicate that it wishes to receive additional DNSSEC RRs (i.e., RRSIG RRs) in the response. Such validating resolvers SHOULD include the DAU, DHU, and/or the N3U option(s) in the OPT RR when sending a query.

4.1.2. Non-validating Stub Resolvers

The DAU, DHU, and N3U EDNS0 options MUST NOT be included by non-validating stub resolvers.

4.2. Recursive Resolvers

4.2.1. Validating Recursive Resolvers

A validating recursive resolver sets the DAU, DHU, and/or N3U option(s) when performing recursion based on its list of algorithms and any DAU, DHU, and/or N3U option lists in the stub client query. When the recursive server receives a query with one or more of the options set, the recursive server MUST set the algorithm list for any outgoing iterative queries for that resolution chain to a union of the stub client's list and the validating recursive resolver's list. For example, if the recursive resolver's algorithm list for the DAU option is (3, 5, 7) and the stub's algorithm list is (7, 8), the final DAU algorithm list would be (3, 5, 7, 8).

If the client included the DO and Checking Disabled (CD) bits, but did not include the DAU, DHU, and/or N3U option(s) in the query, the validating recursive resolver MAY include the option(s) with its own list in full. If one or more of the options are missing, the validating recursive resolver MAY include the missing options with its own list in full.

Validating recursive resolvers MUST NOT set the DAU, DHU, and/or N3U option(s) in the final response to the stub client.

4.2.2. Non-validating Recursive Resolvers

Recursive resolvers that do not do validation MUST copy the DAU, DHU, and/or N3U option(s) seen in received queries as they represent the wishes of the validating downstream resolver that issued the original query.

5. Intermediate System Considerations

Intermediate proxies (see Section 4.4.2 of [RFC5625]) that understand DNS are RECOMMENDED to behave like a comparable recursive resolver when dealing with the DAU, DHU, and N3U options.

6. Server Considerations

When an authoritative server sees the DAU, DHU, and/or N3U option(s) in the OPT meta-RR in a request, the normal algorithm for servicing requests is followed. The options MUST NOT trigger any special processing (e.g., RRSIG filtering in responses) on the server side.

If the options are present but the DO bit is not set, the server does not do any DNSSEC processing, which includes any recording of the option(s).

If the server sees one (or more) of the options set with RESERVED values, the server MAY ignore recoding of those values.

Authoritative servers MUST NOT set the DAU, DHU, and/or N3U option(s) on any responses. These values are only set in queries.

7. Traffic Analysis Considerations

Zone administrators that are planning or are in the process of a cryptographic algorithm rollover operation should monitor DNS query traffic and record the number of queries, the presence of the OPT RR in queries, and the values of the DAU/DHU/N3U option(s) (if present). This monitoring can be used to measure the deployment of client code that implements (and signals) specific algorithms. A description of the techniques used to capture DNS traffic and measure new algorithm adoption is beyond the scope of this document.

Zone administrators that need to comply with changes to their organization's security policy (with regards to cryptographic algorithm use) can use this data to set milestone dates for performing an algorithm rollover. For example, zone administrators can use the data to determine when older algorithms can be phased out without disrupting a significant number of clients. In order to keep this disruption to a minimum, zone administrators should wait to complete an algorithm rollover until a large majority of clients signal that they recognize the new algorithm. This may be in the order of years rather than months.

Note that clients that do not implement these options are likely to be older implementations that would also not implement any newly deployed algorithm.

8. IANA Considerations

The algorithm codes used to identify DNSSEC algorithms, DS RR hash algorithms, and NSEC3 hash algorithms have already been established by IANA. This document does not seek to alter that registry in any way.

IANA has allocated option codes 5, 6, and 7 for the DAU, DHU, and N3U options, respectively, in the "DNS EDNS0 Option Codes (OPT)" registry. The three options have a status of "standard".

9. Security Considerations

This document specifies a way for a client to signal its digital signature and hash algorithm knowledge to a cache or server. It is not meant to be a discussion on algorithm superiority. The signals are optional codes contained in the OPT meta-RR used with EDNS. The goal of these options is to signal new algorithm uptake in client code to allow zone administrators to know when it is possible to complete an algorithm rollover in a DNSSEC-signed zone.

There is a possibility that an eavesdropper or server could infer the validator in use by a client by the presence of the AU options and/or algorithm code list. This information leakage in itself is not very useful to a potential attacker, but it could be used to identify the validator or narrow down the possible validator implementations in use by a client, which could have a known vulnerability that could be exploited by the attacker.

10. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, March 2005.

[RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Resource Records for the DNS Security Extensions", RFC 4034, March 2005.

[RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Protocol Modifications for the DNS Security Extensions", RFC 4035, March 2005.

[RFC5625] Bellis, R., "DNS Proxy Implementation Guidelines", BCP 152, RFC 5625, August 2009.

[RFC6891] Damas, J., Graff, M., and P. Vixie, "Extension Mechanisms for DNS (EDNS(0))", STD 75, RFC 6891, April 2013.

Authors' Addresses

Steve Crocker
Shinkuro Inc.
5110 Edgemoor Lane
Bethesda, MD 20814
USA

EMail: steve@shinkuro.com

Scott Rose
NIST
100 Bureau Dr.
Gaithersburg, MD 20899
USA

Phone: +1-301-975-8439
EMail: scottr.nist@gmail.com