Algorithm Implementation Requirements and Usage Guidance for DNSSEC
draft-wouters-sury-dnsop-algorithm-update-01

Abstract

   The DNSSEC protocol makes use of various cryptographic algorithms in
   order to provide authentication of DNS data and proof of non-
   existence. To ensure interoperability between DNS resolvers and DNS
   authoritative servers, it is necessary to specify a set of algorithm
   implementation requirements and usage guidance to ensure that there
   is at least one algorithm that all implementations support. This
   document defines the current algorithm implementation requirements
   and usage guidance for DNSSEC. This document obsoletes RFC-6944.

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 http://datatracker.ietf.org/drafts/current/.

   Internet-Drafts are draft documents valid for a maximum of six months
   and may be updated, replaced, or obsoleted by other documents at any
   time. It is inappropriate to use Internet-Drafts as reference
   material or to cite them other than as "work in progress."

   This Internet-Draft will expire on September 22, 2016.

Table of Contents

1. Introduction

The DNSSEC signing algorithms are defined by various RFCs, including
[RFC4034], [RFC5155], [RFC5702], [RFC5933], [RFC6605],
[I-D.ietf-curdle-dnskey-ed25519] and [I-D.ietf-curdle-dnskey-ed448].
DNSSEC is used to provide authentication of data. To ensure
interoperability, a set of "mandatory-to-implement" DNSKEY algorithms
are defined. This document obsoletes [RFC6944].

1.1. Updating Algorithm Implementation Requirements and Usage Guidance

The field of cryptography evolves continuously. New stronger
algorithms appear and existing algorithms are found to be less secure
then originally thought. Therefore, algorithm implementation
requirements and usage guidance need to be updated from time to time
to reflect the new reality. The choices for algorithms must be
conservative to minimize the risk of algorithm compromise.

1.2. Updating Algorithm Requirement Levels

The mandatory-to-implement algorithm of tomorrow should already be
available in most implementations of DNSSEC by the time it is made
mandatory. This document attempts to identify and introduce those
algorithms for future mandatory-to-implement status. There is no

guarantee that the algorithms in use today may become mandatory in the future. Published algorithms are continuously subjected to cryptographic attack and may become too weak or could become completely broken before this document is updated.

This document only provides recommendations for the mandatory-to-implement algorithms or algorithms too weak that are recommended not to be implemented. As a result, any algorithm listed at the [DNSKEY-IANA] registry not mentioned in this document MAY be implemented. For clarification and consistency, an algorithm will be set to MAY only when it has been downgraded.

Although this document updates the algorithms to keep the DNSSEC authentication secure over time, it also aims at providing recommendations so that DNSSEC implementations remain interoperable. DNSSEC interoperability is addressed by an incremental introduction or deprecation of algorithms.

It is expected that deprecation of an algorithm is performed gradually. This provides time for various implementations to update their implemented algorithms while remaining interoperable. Unless there are strong security reasons, an algorithm is expected to be downgraded from MUST to MUST- or SHOULD, instead of MUST NOT. Similarly, an algorithm that has not been mentioned as mandatory-to-implement is expected to be introduced with a SHOULD instead of a MUST.

Since the effects of using an unknown DNSKEY algorithm is for the zone to be treated as insecure, it is recommended that algorithms downgraded to SHOULD- or below are no longer used by authoritative nameservers and DNSSEC signers to create new DNSKEY's. This will allow for algorithms to slowly become more unused over time. Once deployment has reached a sufficiently low point these algorithms can finally be marked as MUST NOT so that recursive nameservers can remove support for these algorithms.

Recursive nameservers are encouraged to keep support for all algorithms not marked as MUST NOT.

1.3. Document Audience

The recommendations of this document mostly target DNSSEC
implementers as implementations need to meet both high security
expectations as well as high interoperability between various vendors
and with different versions. Interoperability requires a smooth move
to more secure algorithms. This may differ from a user point of view
that may deploy and configure DNSSEC with only the safest algorithm.
On the other hand, comments and recommendations from this document
are also expected to be useful for such users.

2. Conventions Used in This Document

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].

We define some additional terms here:

SHOULD+   This term means the same as SHOULD. However, it is likely
           that an algorithm marked as SHOULD+ will be promoted at some
           future time to be a MUST.
SHOULD-   This term means the same as SHOULD. However, an algorithm
           marked as SHOULD- may be deprecated to a MAY in a future
           version of this document.
MUST-   This term means the same as MUST. However, we expect at
           some point that this algorithm will no longer be a MUST in a
           future document. Although its status will be determined at
           a later time, it is reasonable to expect that if a future
           revision of a document alters the status of a MUST-
           algorithm, it will remain at least a SHOULD or a SHOULD-.

Table 1

3. Algorithm Selection

3.1. DNSKEY Algorithms

Recommendations for DNSKEY algorithms

Number	Mnemonics	DNSSEC Signing	DNSSEC Validation
1	RSAMD5	MUST NOT	MUST NOT
3	DSA	MUST NOT	MUST NOT
5	RSASHA1	MUST-	MUST-
6	DSA-NSEC3-SHA1	MUST NOT	MUST NOT
7	RSASHA1-NSEC3-SHA1	MUST-	MUST-
8	RSASHA256	MUST	MUST
10	RSASHA512	SHOULD-	MUST

12	ECC-GOST	SHOULD NOT	SHOULD
13	ECDSAP256SHA256	SHOULD+	MUST
14	ECDSAP384SHA384	SHOULD NOT	SHOULD
TBD	ED25519	SHOULD+	SHOULD+
TBD	ED448	SHOULD	SHOULD+

                               Table 2

   RSAMD5 is not widely deployed and there is an industry-wide trend to
   deprecate MD5 usage.

   RSASHA1 and RSASHA1-NSEC3-SHA1 are widely deployed, although zones
   deploying it are recommended to switch to RSASHA256 as there is an
   industry-wide trend to deprecate SHA1 usage. RSASHA1 does not
   support NSEC3. RSASHA1-NSEC3-SHA1 can be used with or without NSEC3.

   DSA and DSA-NSEC3-SHA1 are not widely deployed and vulnerable to
   private key compromise when generating signatures using a weak or
   compromised random number generator.

   RSASHA512 is at the SHOULD level for DNSSEC Signing because it has
   not seen wide deployment, but there are some deployments hence DNSSEC
   Validation MUST implement RSASHA512 to ensure interoperability.

   ECC-GOST is at the SHOULD NOT level because it has not seen wide
   deployment and the algorithm has not seen wide scrutiny in the crypto
   community.

   ECDSAP256SHA256 and ECDSAP384SHA384 provide more strength for
   signature size than RSASHA256 and RSASHA512 variants. It is expected
   ECDSAP256SHA256 will be raised to MUST once it has been deployed more
   widely for DNSSEC Signing. ECDSAP256SHA256 has seen raise in the
   deployment, so it's set to MUST level for DNSSEC Validation.
   ECDSAP384SHA384 offers little over ECDSAP256SHA256 and has not seen
   wide deployment, so it is at the SHOULD NOT level.

   ED25519 and ED448 uses Edwards-curve Digital Security Algorithm
   (EdDSA). There are three main advantages of the EdDSA algorithm: It
   does not require the use of a unique random number for each
   signature, there are no padding or truncation issues as with ECDSA,
   and it is more resilient to side-channel attacks. Hence we expect
   that those algorithms will be raised to MUST once they have been
   deployed more widely.

3.2. DS and CDS Algorithms

Recommendations for Delegation Signer Digest Algorithms. These also
apply to the CDS RRTYPE as specified in [RFC7344]

```
+--------+----------------+------------------+-------------------+
| Number | Mnemonics      | DNSSEC Delegation | DNSSEC Validation |
+--------+----------------+------------------+-------------------+
| 0      | NULL (CDS only) | MUST NOT [*]    | MUST NOT [*]      |
| 1      | SHA-1          | SHOULD NOT       | MUST-             |
| 2      | SHA-256        | MUST             | MUST              |
| 3      | GOST R 34.11-94 | MAY             | SHOULD            |
| 4      | SHA-384        | MAY              | SHOULD+           |
+--------+----------------+------------------+-------------------+
```

[*] - This is a special type of CDS record signaling removal of DS at
             the parent in [I-D.ietf-dnsop-maintain-ds]

                           Table 3

NULL is a special case, see [I-D.ietf-dnsop-maintain-ds]

SHA-1 is in wide use for DS records, but its use is discouraged as it
is an aging algorithm. Users of SHA-1 SHOULD upgrade to SHA-256.

SHA-256 is in wide use and considered strong.

GOST R 34.11-94 is not in wide use. It is still recommended to be
supported in validators so that adoption can increase.

SHA-384 is not in wide use. It is still recommended to be supported
in validators so that adoption can increase.

4. Security Considerations

The security of cryptographic-based systems depends on both the
strength of the cryptographic algorithms chosen and the strength of
the keys used with those algorithms. The security also depends on
the engineering of the protocol used by the system to ensure that
there are no non-cryptographic ways to bypass the security of the
overall system.

   This document concerns itself with the selection of cryptographic
   algorithms for the use of DNSSEC, specifically with the selection of
   "mandatory-to-implement" algorithms. The algorithms identified in
   this document as "MUST implement" or "SHOULD implement" are not known
   to be broken at the current time, and cryptographic research so far
   leads us to believe that they will likely remain secure into the
   foreseeable future. However, this isn't necessarily forever and it
   is expected that new revisions of this document will be issued from
   time to time to reflect the current best practice in this area.

   Retiring an algorithm too soon would result in a signed zone with
   such an algorithm to be downgraded to the equivalent of an unsigned
   zone. Therefore, algorithm deprecation must be done very slowly and
   only after careful consideration and measurements of its use.

5. Operational Considerations

   DNSKEY algorithm rollover in a live zone is a complex process. See
   [RFC6781] and [RFC7583] for guidelines on how to perform algorithm
   rollovers.

6. IANA Considerations

   This document makes no requests of IANA.

7. Acknowledgements

   This document borrows text from RFC 4307 by Jeffrey I. Schiller of
   the Massachusetts Institute of Technology (MIT) and the 4307bis
   document by Yoav Nir, Tero Kivinen, Paul Wouters and Daniel Migault.
   Much of the original text has been copied verbatim.

   We wish to thank Olafur Gudmundsson and Paul Hoffman for their
   imminent feedback.

8. References

8.1. Normative References

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/
              RFC2119, March 1997,
              <http://www.rfc-editor.org/info/rfc2119>.

8.2. Informative References

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

                RFC 4034, DOI 10.17487/RFC4034, March 2005,
                <http://www.rfc-editor.org/info/rfc4034>.

   [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,
                <http://www.rfc-editor.org/info/rfc5155>.

   [RFC5702]   Jansen, J., "Use of SHA-2 Algorithms with RSA in DNSKEY
                and RRSIG Resource Records for DNSSEC", RFC 5702, DOI
                10.17487/RFC5702, October 2009,
                <http://www.rfc-editor.org/info/rfc5702>.

   [RFC5933]   Dolmatov, V., Ed., Chuprina, A., and I. Ustinov, "Use of
                GOST Signature Algorithms in DNSKEY and RRSIG Resource
                Records for DNSSEC", RFC 5933, DOI 10.17487/RFC5933, July
                2010, <http://www.rfc-editor.org/info/rfc5933>.

   [RFC6605]   Hoffman, P. and W. Wijngaards, "Elliptic Curve Digital
                Signature Algorithm (DSA) for DNSSEC", RFC 6605, DOI
                10.17487/RFC6605, April 2012,
                <http://www.rfc-editor.org/info/rfc6605>.

   [RFC6781]   Kolkman, O., Mekking, W., and R. Gieben, "DNSSEC
                Operational Practices, Version 2", RFC 6781, DOI 10.17487/
                RFC6781, December 2012,
                <http://www.rfc-editor.org/info/rfc6781>.

   [RFC6944]   Rose, S., "Applicability Statement: DNS Security (DNSSEC)
                DNSKEY Algorithm Implementation Status", RFC 6944, DOI
                10.17487/RFC6944, April 2013,
                <http://www.rfc-editor.org/info/rfc6944>.

   [RFC7344]   Kumari, W., Gudmundsson, O., and G. Barwood, "Automating
                DNSSEC Delegation Trust Maintenance", RFC 7344, DOI
                10.17487/RFC7344, September 2014,
                <http://www.rfc-editor.org/info/rfc7344>.

   [RFC7583]   Morris, S., Ihren, J., Dickinson, J., and W. Mekking,
                "DNSSEC Key Rollover Timing Considerations", RFC 7583, DOI
                10.17487/RFC7583, October 2015,
                <http://www.rfc-editor.org/info/rfc7583>.

   [I-D.ietf-curdle-dnskey-ed25519]
                Sury, O. and R. Edmonds, "Ed25519 for DNSSEC", draft-ietf-
                curdle-dnskey-ed25519-01 (work in progress), February
                2016.

   [I-D.ietf-curdle-dnskey-ed448]
              Sury, O. and R. Edmonds, "Ed448 for DNSSEC", draft-ietf-
              curdle-dnskey-ed448-00 (work in progress), March 2016.

   [I-D.ietf-dnsop-maintain-ds]
              Gu[eth]mundsson, O. and P. Wouters, "Managing DS records
              from parent via CDS/CDNSKEY", draft-ietf-dnsop-maintain-
              ds-00 (work in progress), December 2015.

   [DNSKEY-IANA]
              , "DNSKEY Algorithms", , <http://www.iana.org/assignments/
              dns-sec-alg-numbers/dns-sec-alg-numbers.xhtml>.

   [DS-IANA] , "Delegation Signer Digest Algorithms", , <http://
              www.iana.org/assignments/ds-rr-types/ds-rr-types.xhtml>.

Authors' Addresses

   Paul Wouters
   Red Hat

   EMail: pwouters@redhat.com

   Ondrej Sury
   CZ.NIC

   EMail: ondrej.sury@nic.cz