Use of SHA-256 in DNSSEC Delegation Signer (DS) Resource Records (RRs)

Status of This Memo

   This document specifies an Internet standards track protocol for the
   Internet community, and requests discussion and suggestions for
   improvements. Please refer to the current edition of the "Internet
   Official Protocol Standards" (STD 1) for the standardization state
   and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Abstract

   This document specifies how to use the SHA-256 digest type in DNS
   Delegation Signer (DS) Resource Records (RRs). DS records, when
   stored in a parent zone, point to DNSKEYs in a child zone.

Table of Contents

1. Introduction

   The DNSSEC [RFC4033] [RFC4034] [RFC4035] DS RR is published in parent
   zones to distribute a cryptographic digest of one key in a child's
   DNSKEY RRset. The DS RRset is signed by at least one of the parent
   zone's private zone data signing keys for each algorithm in use by
   the parent. Each signature is published in an RRSIG resource record,
   owned by the same domain as the DS RRset, with a type covered of DS.

   In this document, the key words "MUST", "MUST NOT", "REQUIRED",
   "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY",
   and "OPTIONAL" are to be interpreted as described in [RFC2119].

2. Implementing the SHA-256 Algorithm for DS Record Support

   This document specifies that the digest type code 2 has been assigned
   to SHA-256 [SHA256] [SHA256CODE] for use within DS records. The
   results of the digest algorithm MUST NOT be truncated, and the entire
   32 byte digest result is to be published in the DS record.

2.1. DS Record Field Values

   Using the SHA-256 digest algorithm within a DS record will make use
   of the following DS-record fields:

   Digest type: 2

   Digest: A SHA-256 bit digest value calculated by using the following
      formula ("|" denotes concatenation). The resulting value is not
      truncated, and the entire 32 byte result is to be used in the
      resulting DS record and related calculations.

         digest = SHA_256(DNSKEY owner name | DNSKEY RDATA)

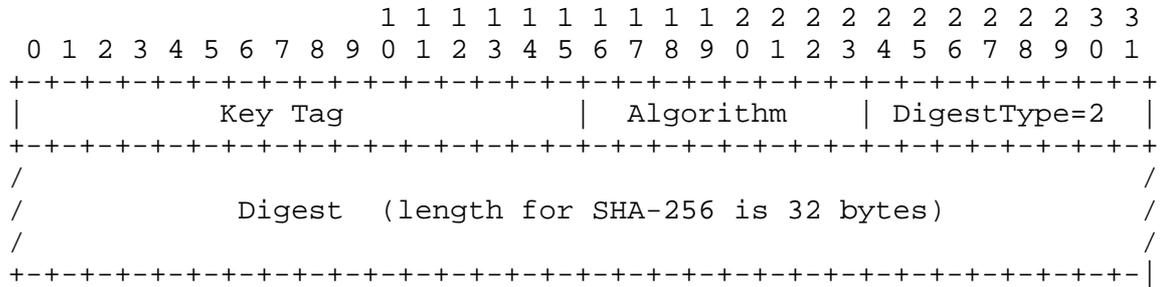   where DNSKEY RDATA is defined by [RFC4034] as:

         DNSKEY RDATA = Flags | Protocol | Algorithm | Public Key

   The Key Tag field and Algorithm fields remain unchanged by this
   document and are specified in the [RFC4034] specification.

2.2. DS Record with SHA-256 Wire Format

   The resulting on-the-wire format for the resulting DS record will be
   as follows:

```
                          1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     |           Key Tag             |  Algorithm    | DigestType=2  |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
     /                                                               /
     /            Digest  (length for SHA-256 is 32 bytes)           /
     /                                                               /
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-|
```

2.3. Example DS Record Using SHA-256

   The following is an example DNSKEY and matching DS record. This
   DNSKEY record comes from the example DNSKEY/DS records found in
   section 5.4 of [RFC4034].

   The DNSKEY record:

   dskey.example.com. 86400 IN DNSKEY 256 3 5 (AQOeiiR0GOMYkDshWoSKz9Xz
                                                fwJr1AYtsmx3TGkJaNXVbfi/
                                                2pHm822aJ5iI9BMzNXxeYCmZ
                                                DRD99WYwYqUSdjMmmAphXdvx
                                                egXd/M5+X7OrzKBaMbCVdFLU
                                                Uh6DhweJBjEVv5f2wwjM9Xzc
                                                nOf+EPbtG9DMBmADjFDc2w/r
                                                ljwvFw==
) ; key id = 60485

   The resulting DS record covering the above DNSKEY record using a
   SHA-256 digest:

   dskey.example.com. 86400 IN DS 60485 5 2   (D4B7D520E7BB5F0F67674A0C
                                                CEB1E3E0614B93C4F9E99B83
                                                83F6A1E4469DA50A)

3. Implementation Requirements

   Implementations MUST support the use of the SHA-256 algorithm in DS
   RRs. Validator implementations SHOULD ignore DS RRs containing SHA-1
   digests if DS RRs with SHA-256 digests are present in the DS RRset.

4. Deployment Considerations

   If a validator does not support the SHA-256 digest type and no other
   DS RR exists in a zone's DS RRset with a supported digest type, then
   the validator has no supported authentication path leading from the
   parent to the child. The resolver should treat this case as it would
   the case of an authenticated NSEC RRset proving that no DS RRset
   exists, as described in [RFC4035], Section 5.2.

   Because zone administrators cannot control the deployment speed of
   support for SHA-256 in validators that may be referencing any of
   their zones, zone operators should consider deploying both SHA-1 and
   SHA-256 based DS records. This should be done for every DNSKEY for
   which DS records are being generated. Whether to make use of both
   digest types and for how long is a policy decision that extends
   beyond the scope of this document.

5. IANA Considerations

   Only one IANA action is required by this document:

   The Digest Type to be used for supporting SHA-256 within DS records
   has been assigned by IANA.

   At the time of this writing, the current digest types assigned for
   use in DS records are as follows:

       VALUE   Digest Type   Status
         0   Reserved   -
         1   SHA-1   MANDATORY
         2   SHA-256   MANDATORY
       3-255   Unassigned   -

6. Security Considerations

6.1. Potential Digest Type Downgrade Attacks

   A downgrade attack from a stronger digest type to a weaker one is
   possible if all of the following are true:

   o A zone includes multiple DS records for a given child's DNSKEY,
      each of which uses a different digest type.

   o A validator accepts a weaker digest even if a stronger one is
      present but invalid.

For example, if the following conditions are all true:

o Both SHA-1 and SHA-256 based digests are published in DS records
   within a parent zone for a given child zone's DNSKEY.

o The DS record with the SHA-1 digest matches the digest computed
   using the child zone's DNSKEY.

o The DS record with the SHA-256 digest fails to match the digest
   computed using the child zone's DNSKEY.

Then, if the validator accepts the above situation as secure, then
this can be used as a downgrade attack since the stronger SHA-256
digest is ignored.

6.2. SHA-1 vs. SHA-256 Considerations for DS Records

Users of DNSSEC are encouraged to deploy SHA-256 as soon as software
implementations allow for it. SHA-256 is widely believed to be more
resilient to attack than SHA-1, and confidence in SHA-1's strength is
being eroded by recently announced attacks. Regardless of whether
the attacks on SHA-1 will affect DNSSEC, it is believed (at the time
of this writing) that SHA-256 is the better choice for use in DS
records.

At the time of this publication, the SHA-256 digest algorithm is
considered sufficiently strong for the immediate future. It is also
considered sufficient for use in DNSSEC DS RRs for the immediate
future. However, future published attacks may weaken the usability
of this algorithm within the DS RRs. It is beyond the scope of this
document to speculate extensively on the cryptographic strength of
the SHA-256 digest algorithm.

Likewise, it is also beyond the scope of this document to specify
whether or for how long SHA-1 based DS records should be
simultaneously published alongside SHA-256 based DS records.

7. Acknowledgements

This document is a minor extension to the existing DNSSEC documents
and those authors are gratefully appreciated for the hard work that
went into the base documents.

The following people contributed to portions of this document in some
fashion: Mark Andrews, Roy Arends, Olafur Gudmundsson, Paul Hoffman,
Olaf M. Kolkman, Edward Lewis, Scott Rose, Stuart E. Schechter, Sam
Weiler.

8. References

8.1. 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.

   [SHA256]   National Institute of Standards and Technology, "Secure
                  Hash Algorithm. NIST FIPS 180-2", August 2002.

8.2. Informative References

   [SHA256CODE] Eastlake, D., "US Secure Hash Algorithms (SHA)", Work in
                  Progress.

Author's Address

   Wes Hardaker
   Sparta
   P.O. Box 382
   Davis, CA 95617
   USA

   EMail: hardaker@tislabs.com

Full Copyright Statement

   Copyright (C) The Internet Society (2006).

   This document is subject to the rights, licenses and restrictions
   contained in BCP 78, and except as set forth therein, the authors
   retain all their rights.

   This document and the information contained herein are provided on an
   "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS
   OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET
   ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED,
   INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE
   INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED
   WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

   The IETF takes no position regarding the validity or scope of any
   Intellectual Property Rights or other rights that might be claimed to
   pertain to the implementation or use of the technology described in
   this document or the extent to which any license under such rights
   might or might not be available; nor does it represent that it has
   made any independent effort to identify any such rights. Information
   on the procedures with respect to rights in RFC documents can be
   found in BCP 78 and BCP 79.

   Copies of IPR disclosures made to the IETF Secretariat and any
   assurances of licenses to be made available, or the result of an
   attempt made to obtain a general license or permission for the use of
   such proprietary rights by implementers or users of this
   specification can be obtained from the IETF on-line IPR repository at
   http://www.ietf.org/ipr.

   The IETF invites any interested party to bring to its attention any
   copyrights, patents or patent applications, or other proprietary
   rights that may cover technology that may be required to implement
   this standard. Please address the information to the IETF at
   ietf-ipr@ietf.org.

Acknowledgement

   Funding for the RFC Editor function is provided by the IETF
   Administrative Support Activity (IASA).