Network Working Group
Request for Comments: 3110
Obsoletes: 2537
Category: Standards Track

D. Eastlake 3rd
Motorola

May 2001

RSA/SHA-1 SIGs and RSA KEYs in the Domain Name System (DNS)

Status of this Memo

Copyright Notice

Abstract

This document describes how to produce RSA/SHA1 SIG resource records (RRs) in Section 3 and, so as to completely replace RFC 2537, describes how to produce RSA KEY RRs in Section 2.

Since the adoption of a Proposed Standard for RSA signatures in the DNS (Domain Name Space), advances in hashing have been made. A new DNS signature algorithm is defined to make these advances available in SIG RRs. The use of the previously specified weaker mechanism is deprecated. The algorithm number of the RSA KEY RR is changed to correspond to this new SIG algorithm. No other changes are made to DNS security.

Acknowledgements

Material and comments from the following have been incorporated and are gratefully acknowledged:

Olafur Gudmundsson

The IESG

Charlie Kaufman

Steve Wang

Table of Contents

1. Introduction

The Domain Name System (DNS) is the global hierarchical replicated distributed database system for Internet addressing, mail proxy, and other information [RFC1034, 1035, etc.]. The DNS has been extended to include digital signatures and cryptographic keys as described in [RFC2535]. Thus the DNS can now be secured and used for secure key distribution.

Familiarity with the RSA and SHA-1 algorithms is assumed [Schneier, FIP180] in this document.

RFC 2537 described how to store RSA keys and RSA/MD5 based signatures in the DNS. However, since the adoption of RFC 2537, continued cryptographic research has revealed hints of weakness in the MD5 [RFC1321] algorithm used in RFC 2537. The SHA1 Secure Hash Algorithm [FIP180], which produces a larger hash, has been developed. By now there has been sufficient experience with SHA1 that it is generally acknowledged to be stronger than MD5. While this stronger hash is probably not needed today in most secure DNS zones, critical zones such a root, most top level domains, and some second and third level domains, are sufficiently valuable targets that it would be negligent not to provide what are generally agreed to be stronger mechanisms. Furthermore, future advances in cryptanalysis and/or computer speeds may require a stronger hash everywhere. In addition, the additional computation required by SHA1 above that required by MD5 is insignificant compared with the computational effort required by the RSA modular exponentiation.

This document describes how to produce RSA/SHA1 SIG RRs in Section 3 and, so as to completely replace RFC 2537, describes how to produce RSA KEY RRs in Section 2.

Implementation of the RSA algorithm in DNS with SHA1 is MANDATORY for DNSSEC. The generation of RSA/MD5 SIG RRs as described in RFC 2537 is NOT RECOMMENDED.

The key words "MUST", "REQUIRED", "SHOULD", "RECOMMENDED", "NOT RECOMMENDED", and "MAY" in this document are to be interpreted as described in RFC 2119.

2. RSA Public KEY Resource Records

RSA public keys are stored in the DNS as KEY RRs using algorithm number 5 [RFC2535]. The structure of the algorithm specific portion of the RDATA part of such RRs is as shown below.

Field	Size
exponent length	1 or 3 octets (see text)
exponent	as specified by length field
modulus	remaining space

For interoperability, the exponent and modulus are each limited to 4096 bits in length. The public key exponent is a variable length unsigned integer. Its length in octets is represented as one octet if it is in the range of 1 to 255 and by a zero octet followed by a two octet unsigned length if it is longer than 255 bytes. The public key modulus field is a multiprecision unsigned integer. The length of the modulus can be determined from the RDLENGTH and the preceding RDATA fields including the exponent. Leading zero octets are prohibited in the exponent and modulus.

Note: KEY RRs for use with RSA/SHA1 DNS signatures MUST use this algorithm number (rather than the algorithm number specified in the obsoleted RFC 2537).

Note: This changes the algorithm number for RSA KEY RRs to be the same as the new algorithm number for RSA/SHA1 SIGs.

3. RSA/SHA1 SIG Resource Records

RSA/SHA1 signatures are stored in the DNS using SIG resource records (RRs) with algorithm number 5.

The signature portion of the SIG RR RDATA area, when using the RSA/SHA1 algorithm, is calculated as shown below. The data signed is determined as specified in RFC 2535. See RFC 2535 for fields in the SIG RR RDATA which precede the signature itself.

```
hash = SHA1 ( data )

signature = ( 01 | FF* | 00 | prefix | hash ) ** e (mod n)
```

where SHA1 is the message digest algorithm documented in [FIP180], "|" is concatenation, "e" is the private key exponent of the signer, and "n" is the modulus of the signer's public key. 01, FF, and 00 are fixed octets of the corresponding hexadecimal value. "prefix" is the ASN.1 BER SHA1 algorithm designator prefix required in PKCS1 [RFC2437], that is,

hex 30 21 30 09 06 05 2B 0E 03 02 1A 05 00 04 14

This prefix is included to make it easier to use standard cryptographic libraries. The FF octet MUST be repeated the maximum number of times such that the value of the quantity being exponentiated is one octet shorter than the value of n.

(The above specifications are identical to the corresponding parts of Public Key Cryptographic Standard #1 [RFC2437].)

The size of "n", including most and least significant bits (which will be 1) MUST be not less than 512 bits and not more than 4096 bits. "n" and "e" SHOULD be chosen such that the public exponent is small. These are protocol limits. For a discussion of key size see RFC 2541.

Leading zero bytes are permitted in the RSA/SHA1 algorithm signature.

4. Performance Considerations

General signature generation speeds are roughly the same for RSA and DSA [RFC2536]. With sufficient pre-computation, signature generation with DSA is faster than RSA. Key generation is also faster for DSA. However, signature verification is an order of magnitude slower with DSA when the RSA public exponent is chosen to be small as is recommended for KEY RRs used in domain name system (DNS) data authentication.

A public exponent of 3 minimizes the effort needed to verify a signature. Use of 3 as the public exponent is weak for confidentiality uses since, if the same data can be collected encrypted under three different keys with an exponent of 3 then, using the Chinese Remainder Theorem [NETSEC], the original plain text can be easily recovered. If a key is known to be used only for authentication, as is the case with DNSSEC, then an exponent of 3 is acceptable. However other applications in the future may wish to leverage DNS distributed keys for applications that do require confidentiality. For keys which might have such other uses, a more conservative choice would be 65537 (F4, the fourth fermat number).

Current DNS implementations are optimized for small transfers, typically less than 512 bytes including DNS overhead. Larger transfers will perform correctly and extensions have been standardized [RFC2671] to make larger transfers more efficient, it is still advisable at this time to make reasonable efforts to minimize the size of KEY RR sets stored within the DNS consistent with adequate security. Keep in mind that in a secure zone, at least one authenticating SIG RR will also be returned.

5. IANA Considerations

The DNSSEC algorithm number 5 is allocated for RSA/SHA1 SIG RRs and RSA KEY RRs.

6. Security Considerations

Many of the general security considerations in RFC 2535 apply. Keys retrieved from the DNS should not be trusted unless (1) they have been securely obtained from a secure resolver or independently verified by the user and (2) this secure resolver and secure obtainment or independent verification conform to security policies acceptable to the user. As with all cryptographic algorithms, evaluating the necessary strength of the key is essential and dependent on local policy. For particularly critical applications, implementers are encouraged to consider the range of available algorithms and key sizes. See also RFC 2541, "DNS Security Operational Considerations".

References

[FIP180] U.S. Department of Commerce, "Secure Hash Standard", FIPS PUB 180-1, 17 Apr 1995.

[NETSEC] Network Security: PRIVATE Communications in a PUBLIC World, Charlie Kaufman, Radia Perlman, & Mike Speciner, Prentice Hall Series in Computer Networking and Distributed Communications, 1995.

[RFC1034] Mockapetris, P., "Domain Names - Concepts and Facilities", STD 13, RFC 1034, November 1987.

[RFC1035] Mockapetris, P., "Domain Names - Implementation and Specification", STD 13, RFC 1035, November 1987.

[RFC1321] Rivest, R., "The MD5 Message-Digest Algorithm", RFC 1321, April 1992.

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

[RFC2437] Kaliski, B. and J. Staddon, "PKCS #1: RSA Cryptography Specifications Version 2.0", RFC 2437, October 1998.

[RFC2535] Eastlake, D., "Domain Name System Security Extensions", RFC 2535, March 1999.

[RFC2536] Eastlake, D., "DSA KEYs and SIGs in the Domain Name System (DNS)", RFC 2536, March 1999.

[RFC2537] Eastlake, D., "RSA/MD5 KEYs and SIGs in the Domain Name System (DNS)", RFC 2537, March 1999.

[RFC2541] Eastlake, D., "DNS Security Operational Considerations", RFC 2541, March 1999.

[RFC2671] Vixie, P., "Extension Mechanisms for DNS (EDNS0)", RFC 2671, August 1999.

[Schneier] Bruce Schneier, "Applied Cryptography Second Edition: protocols, algorithms, and source code in C", 1996, John Wiley and Sons, ISBN 0-471-11709-9.

Author's Address

Donald E. Eastlake 3rd
Motorola
155 Beaver Street
Milford, MA 01757 USA

Phone: +1-508-261-5434 (w)
+1-508-634-2066 (h)
Fax +1-508-261-4777 (w)
EMail: Donald.Eastlake@motorola.com

Full Copyright Statement

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.