Indirect KEY RRs in the Domain Name System

Donald E. Eastlake 3rd

Status of This Document

This draft, file name draft-ietf-dnssec-indirect-key-01.txt, is intended to be become a Proposed Standard RFC. Distribution of this document is unlimited. Comments should be sent to the DNSSEC mailing list <dns-security@tis.com> or to the author.

This document is an Internet-Draft. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months. Internet-Drafts may be updated, replaced, or obsoleted by other documents at any time. It is not appropriate to use Internet-Drafts as reference material or to cite them other than as a ``working draft'' or ``work in progress.''

To learn the current status of any Internet-Draft, please check the 1id-abstracts.txt listing contained in the Internet-Drafts Shadow Directories on ds.internic.net (East USA), ftp.isi.edu (West USA), nic.nordu.net (North Europe), ftp.nis.garr.it (South Europe), munnari.oz.au (Pacific Rim), or ftp.is.co.za (Africa).

Abstract

RFC 2065 defines a means for storing cryptogrpahic public keys in the Domain Name System. An additional code point is defined for the KEY resource record (RR) algorithm field to indicate that the key itself is not stored in the KEY RR but is pointed to by the KEY RR. Encodings to indicate different types of key and pointer formats are specified.

Table of Contents

1. Introduction

The Domain Name System (DNS) security extensions [RFC 2065] provide for the general storage of public keys in the domain name system via the KEY resource record (RR). These KEY RRs are used in support of DNS security and may be used to support other security protocols. KEY RRs can be associated with users, zones, and hosts or other end entities named in the DNS.

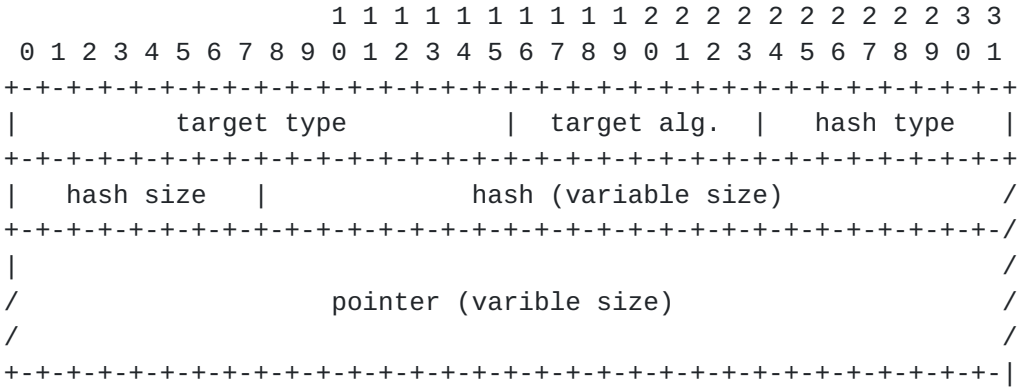
For reasons given below, in many cases it will be desireable to store a key or keys elsewhere and merely point to it from the KEY RR. Indirect key storage makes it possible to point to a key service via a URL, to have a compact pointer to a larger key or set of keys, to point to a certificate either inside DNS [see draft-ietf-dnssec-certs-*.txt] or outside the DNS, and where appropriate, to store a key or key set applicable to many DNS entries in some place and point to it from those entries.

However, to simplify DNSSEC implementation, this technique MUST NOT be used for KEY RRs used in for verification in DNSSEC.

2. The Indirect KEY RR Algorithm

Domain Name System (DNS) KEY Resource Record (RR) [RFC 2065] algorithm number 252 is defined as the indirect key algorithm. This algorithm MAY NOT be used for zone keys in support of DNS security. All KEYs used in DNSSEC validation must be stored directly in the DNS.

When the algorithm byte of a KEY RR has thae value 252, the "public key" portion of the RR is formated as follows:



2.1 The Target Type Field

Target type specifies the type of the key containing data being pointed at.

Target types 0 and 65535 are reserved.

Target type 1 indicates that the pointer is a domain name from which KEY RRs [RFC 2065] should be retrieved. Name compression in the pointer field is prohibited.

Target type 2 indicates that the pointer is a null terminated character string which is a URL [RFC 1738]. For exisiting data transfer URL schemes, such as ftp, http, shttp, etc., the data is the same as the public key portion of a KEY RR. (New URL schmes may be defined which return multiple keys.)

Target type 2 indicates that the pointer is a domain name from which CERT RRs [draft-ietf-dnssec-certs-*.txt] should be retrieved. Name compression in the pointer field is prohibiited.

Target type 3 indicates that the pointer is a null terminated character string which is a URL [RFC 1738]. For exisiting data transfer URL schemes, such as ftp, http, shttp, etc., the data is the

same as the entire RDATA portion of a CERT RR [draft-ietf-dnssec-

certs-*.txt]. (New URL schmes may be defined which return multiple such data blocks.)

Target type 4 indicates that the pointer is a null terminated character string which is a URL [RFC 1738]. For exisiting data transfer URL schemes, such as ftp, http, shttp, etc., the data is a PKCS#1 format key. (New URL schmes may be defined which return multiple keys.)

The target types 5 through 255 are available for assignment by IANA.

Target type 256 through 511 (i.e., 256 + n) indicate that the pointer is a null terminated character string which is a URL [RFC 1738]. For exisiting data transfer URL schemes, such as ftp, http, shttp, etc., the data is a certificate of the type indicated by a CERT RR [draft-ietf-dnssec-certs-*.txt] certificate type of n. That is, target types 257, 258, and 259 are PKIX, SPKI, and PGP certificates and target types 509 and 510 are URL and OID private certificate types. (New URL schmes may be defined which return multiple such certificates.)

Target types 512 through 65534 are available for assignment by IANA.

2.2 The Target Algorithm Field

The algorithm field is as defined in RFC 2065. if non-zero, it specifies the algorithm type of the target key or keys pointed. If zero, it does not specify what algorithm the target key or keys apply to.

2.3 The Hash Fields

If the indirecting KEY RR is retrieved from an appropriately secure DNS zone with a resolver implementing DNS security, then there would be a high level of confidence in the entire value of the KEY RR including any direct keys. This may or may not be true of any indirect key pointed to. If that key is embodied in a certificate or retrieved via a secure protocol such as SHTTP, it may also be secure. But an indirecting KEY RR could, for example, simply have an FTP URL pointing to a binary key stored elsewhere, the retrieval of which would not be secure.

The hash option in algorithm 252 KEY RRs provides a means of extending the security of the indirecting KEY RR to the actual key material pointed at. By incldуing a hash in a secure indirecting RR,

this secure hash can be checked against the hash of the actual keying

material

Type	Hash Algorithm
0	indicates no hash present
1	MD5 [RFC 1321]
2	SHA-1
3	RIPEMD
4-254	available for assignment
255	reserved

The hash size field is an unsigned octet count of the hash size. For some hash algorithms it may be fixed by the algorithm choice but this will not always be the case. For example, hash size is used to distinguish between RIPEMD-128 (16 octets) and RIPEMD-160 (20 octets). If the hash algorithm is 0, the hash size MUST be zero and no hash octets are present.

The hash field itself is variable size with its length specified by the hash size field.

3. Performance Considerations

With current public key technology, an indirect key will sometimes be shorter than the keying material it points at. This may improve DNS permformace in the retrieval of the initial KEY RR. However, an additional retrieval step then needs to be done to get the actualy keying material which must be added to the overall time to get the public key.

4. Security Considerations

The indirecting step of using an indirect KEY RR adds complexity and additional steps where security could go wrong. If the indirect key RR was retrieved from a zone that was insecure for the resolver, you have no security. If the indirect key RR, although secure itself, point to a key which can not be securely retrieved and is not validatated by a secure hash in the indirect key RR, you have no security.

References

PKCS#1

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

RFC 1035 - P. Mockapetris, "Domain Names - Implementation and Specifications", STD 13, November 1987.

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

RFC 1738 - T. Berners-Lee, L. Masinter & M. McCahill, "Uniform Resource Locators (URL)", December 1994.

RFC 2065 - D. Eastlake, C. Kaufman, "Domain Name System Security Extensions", 01/03/1997.

draft-ietf-dnssec-certs-*.txt

Author's Address

Donald E. Eastlake 3rd
CyberCash, Inc.
318 Acton Street
Carlisle, MA 01741 USA

Telephone: +1 978 287 4877
+1 703 620-4200 (main office, Reston, VA)
FAX: +1 978 371 7148
EMail: dee@cybercash.com

Expiration and File Name

This draft expires May 1998.

Its file name is draft-ietf-dnssec-indirect-key-01.txt.