DNSIND WG
INTERNET DRAFT
Category: I-D

Edward Lewis
NAI Labs
Jerry Scharf
ISC
Olafur Gudmundsson
NAI Labs
June 25, 1999

The SEC Resource Record
<draft-ietf-dnsind-sec-rr-00.txt>

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026.

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 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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Comments should be sent to the authors or the DNSIND WG mailing list namedroppers@internic.net.

This draft expires on December 25, 1999.

Abstract

A new DNS reseource record, the SECurity RR, is defined to address concerns about the parent zone's holding of the child zone's KEY RR set. These concerns are addressed in a manner that retains the information needed by a secure resolver when asking a parent zone about the child zone. This proposal updates RFC 2535 and RFC 2181.

1. Introduction

DNS security extensions require a signed zone to hold KEY RR sets for each of its delegations. This requirement has four negative implications for the top level domains, which, for the most part,

consist of delegation points. (These issues also impact other delegating zones, these problems are not unique to the TLDs.) Addressing these concerns by removing the requirement for the KEY RR in the parent has an adverse effect on secure resolution of DNS

signatures. A new DNS reseource record, the SECurity RR, is defined to address these concerns.

The Zone Key Referral, described in another draft by the same authors, is one proposed response to the concerns about parent's holding child keys. However, that proposal has two drawbacks. One, it results in two KEY RR sets at a delegation, one in the parent and one in the child, which differ. It also does not address the expression of security parameters, such as whether or not the child zone uses the NXT record (which is currently mandatory).

This document will begin by repeating the arguments against the holding of keys at the parent as presented in the Zone Key Referral. The document will then present the need for information about the child to be held in parent. Following this, the SEC RR will be defined, its master file representation discussed, and implications on name servers.

(Editorial note. Sections 1.1 through 1.5 are copied nearly verbatim from the Zone Key Referral so that retirement of that draft will not cause a problem.)

1.1 Reasons for removing the KEY data from the parent

There are a number of different reasons for the removal of the KEY RR from the parent. Reasons include:

- o the performance impact that holding keys has on name servers
- o the problem of updating a widely delegated parent zone on demand
- o statements in RFC 2181 on authoritative data at delegations
- o perceived liability of the operator of a name server or registry

1.2 Performance Issues

A sample zone will be used to illustrate the problem. The example will part from reality mostly in the length of zone names, which changes the size of the owner and resource record data fields.


```
        # $ORIGIN test.
        # @         IN SOA   <SOA data>
        #           IN SIG   SOA <by test.>
```

```
#           IN KEY   <1024 bit zone key>
#           IN SIG   KEY <by test.>
#           IN SIG   KEY <by .>
#           IN NS    ns.test.
#           IN SIG   NS <by test.>
#           IN NXT   my-org.test. NS SOA SIG KEY NXT
#           IN SIG   NXT <by test.>
#
# my-org    IN KEY   <1024 bit zone key>
#           IN KEY   <1024 bit zone key>
#           IN SIG   KEY <by test.>
#           IN NS    ns1.my-org.test.
#           IN NS    ns2.my-org.test.
#           IN NXT   that-org.test. NS SIG KEY NXT
#           IN SIG   NXT <by test.>
#
# that-org  IN KEY   0xC100 3 255
#           IN SIG   KEY <by test.>
#           IN NS    ns1.that-org.test.
#           IN NS    ns2.that-org.test.
#           IN NXT   test. NS SIG KEY NXT
#           IN SIG   NXT <by test.>
```

In this zone file, "my-org" is a delegation point of interest with two registered public keys. Presumably, one key is for signatures generated currently and the other is for still living and valid but older signatures. "that-org" is another delegation point, with a NULL key. This signifies that this zone is unsecured.

To analyze the performance impact of the storing of keys, the number of bytes used to represent the RRs in the procotol format is used. The actual number of bytes stored will likely be higher, accounting for data structure overhead and alignment. The actual number of bytes transferred will be lower due to DNS name compression.

The number of bytes for my-org's two 1024-bit keys, two NS records, NXT and the associated signatures is 526. (1024 bit RSA/MD5 keys were used for the calculation.) The bytes needed for that-org (with the NULL key) is 346. Currently, there are close to 2 million entries in com., so if we take my-org as a typical domain, over 1GB on memory will be needed for com. The zone keys used in the example are set to **1024 bits.** This number may range from as low as 512 bits upwards to over 3000 bits.

The increased size of the data held for the zone cuts will have two impacts at the transport and below layers. Bandwidth beyond that currently needed will be used to carry the KEY records. The inclusion of all of the child's keys will also push DNS over the UDP size limit and start using TCP - which could cause critical problems for current

heavily used name servers, like the root and TLD name servers. EDNS0 [RFC-to-be] addresses expansion of UDP message size, which alleviates this problem.

Another impact, not illustrated by the example, is the frequency of updates. If each time a public key for my-org is added or deleted, the SOA serial number will have to increase, and the SOA signed again. If an average zone changes the contents of its key RR set once per month, there will be on average 45 updates per minute in a zone of 2 million delegations. (This estimate does not address the fact that signatures also expire, requiring a new signing of the zone periodically.)

1.3 Security Incident Recovery (w/ respect to keys only)

Once a zone administrator is alerted that any key's private counterpart has been discovered (exposed), the first action to be taken is to stop advertising the public key in DNS. This doesn't end the availability of the key - it will be residing in caches and given in answers from those caches - but is the closest action resembling revokation available in DNS.

Stopping the advertisement in the zone's name servers is as quick as altering the master file and restarting the name server. Having to do this in two places will will only delay the time until the recovery is complete.

For example, a registrar of a top level domain has decided to update its zone only on Mondays and Fridays due to the size of the zone. A customer/delegatee is the victim of a break in, in which one of the items taken is the file of private keys used to sign DNS data. If this occurs on a Tuesday, the thief has until Friday to use the keys before they disappear from the DNS, even if the child stops publishing them immediately.

If the public key set is in the parent zone, and the parent zone is not able to make the change quickly, the public key cannot be revoked quickly. If the parent only refers to there being a key at the child zone, then the child has the agility to change the keys - even issue a NULL key, which will force all signatures in the zone to become suspect.

1.4 DNS Clarifications

RFC 2181, section 6, clarifies the status of data appearing at a zone cut. Data at a zone cut is served authoritatively from the servers listed in the NS set present at the zone cut. The data is not (necessarily) served authoritatively from the parent. (The exception is in servers handling both the parent and child zone.)

Section 6 also mentions that there are two exceptions created by DNSSEC, the NXT single record set and the KEY set. This proposal addresses the exception relating to the KEY set, by removing the set

from the parent. The SEC RR is introduced and belongs in the parent zone, there is no counterpart in the child (at the apex).

1.5 Liability

Liability is a legal concept, so it is not wise to attempt an engineering solution to it. However, the perceived liability incurred in using DNSSEC by registrars may prevent the adoption of DNSSEC. Hence DNSSEC should be engineered in such a way to address the concern.

One source of liability is the notion that by advertising a public key for a child zone, a parent zone is providing a service of security. With that comes responsibility. By having the parent merely indicate that a child has a key (or has no key), the parent is providing less in the way of security. If the parent is wrong, the potential loss is less. Instead of falsely authenticated data, configuration errors will be apparent to the resolving client.

Whether or not the KEY RR remains advertised in the parent zone or the SEC RR is in place, the parent zone administrators still have to adhere to proper key handling practices, which are being documented in DNSOP draft. In particular, the parent has to be sure that the keys it is signing for a child have been submitted by the true administrator of the the child zone, and not submitted by an imposter.

1.6 The needs of the resolver

Now that the reasons for removing the child's keys from the parent zone have been presented, reasons why something must take their place are presented. A "secure" resolver is a DNS resolver that receives an answer and, if a signature arrives, verifies the signature. Most often, this operation will happen in resolvers that are part of name servers, as opposed to general purpose hosts.

The first step in authenticating a DNS response is to see if the data is accompanied by a signature. There are five possible outcomes. Three results are not desirable, a signature may arrive but shouldn't, no signature arrives but should, or a signature arrives but uses the wrong cryptographic algorthm Two outcomes can be considered successful, a signature arriving with the correct algorithm or no signature arrives and shouldn't. (There is one other case - a signature generated with an inappropriate key - which is a matter beyond the scope of this draft.)

Since the resolver can not instantly know whether a signature is expected, the resolver must start a discovery process. This process can be done by the resolver querying zones between the root and the desired domain for information about the next successive zones. (Optimizing this search is not presented here.) For this search to be successful, the parent must hold something that indicates the status of the child's security, so the resolver may search with certainty. While refraining from using the word "policy" to describe the data, the phrase "security parameters" is used.

The security parameters of a zone are not entirely defined yet, and will remain open until a critical mass of operations experience is gained. Initially, the following information is known to be needed.

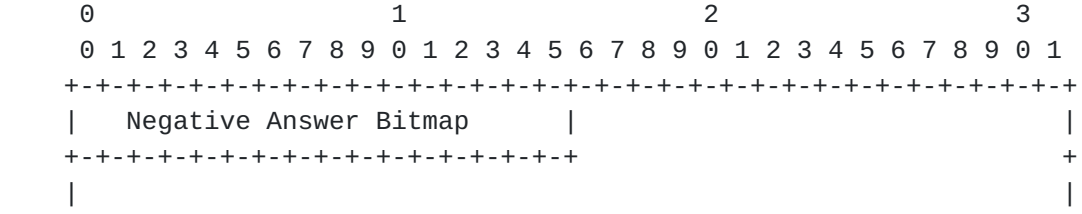
The set of algorithms in use by the zone.
KEY RRs and SIG RRs have protocol fields indicating how the key is made. For now, two are in distribution, a value of 1 for RSA/MD5 and 3 **for DSA.** Unfortunately, the value are numeric in 8 bits, so a bitmap representation cannot be used.

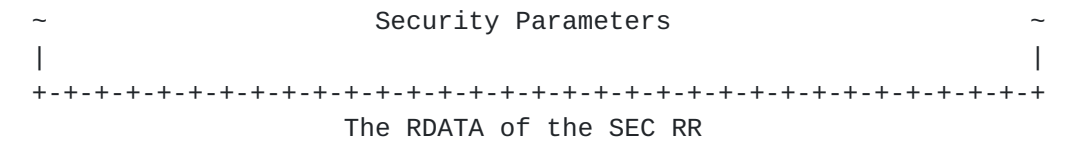
The mechanism for negative answers.
Currently, the NXT is mandatory, liked by some administrators and disliked by other administrators. NXTs cannot be made optional, doing so makes them obsolete. (An attacker can make the responses look like a zone doesn't use NXTs, even if the zone does.) If the choice of NXT or no NXT can be securely indicated, then this is solved. There have been discussions on alternatives to the NXT record. By allowing a zone to indicate the style of negative answer in use, alternatives can be installed in experimental zones.

Signature policy.
This is an untested issue. Expressing a policy, such as whether multiple algorithms are used, whether verification of one signature needed or all signatures, etc., has not been fully studied.

2. The SEC RR

The SEC RR is a record that describes the DNS security parameters of the owner. The owner MUST also have an NS RR set, i.e., the owner MUST be a cut point. A signed zone MUST have a SEC RR set for each delegation point.





The RDATA of the SEC RR

The SEC RR RDATA contains two data fields. One is a 16-bit field acting as a bitmap to indicate the means used to signify a negative answer. The other field is an unbounded field of option-value pairs indicating other salient settings for the zone. The latter field is not padded to any particular byte boundary.

The SEC RR is answered authoritatively from the parent zone, and is signed by the parent. A properly configured delegation point in the parent would have just an SEC RR, records used for negative answering, and a glue NS set. The corresponding point in the child (the zone's apex) would have the SOA, KEY set, NS records, negative answer

records, and other desired and legal RR sets. SIG RR's appear in both the parent and child side of the delegation.

There is no other special processing of the SEC RR set. It is used in a reply as an answer when it is the subject of a direct query (QTYPE IS SEC) or when a QTYPE=ANY reaches the delegating zone. If a name server is authoritative for both the parent and child, the SEC is included in the ANY reply for the delegation point.

(Editorial note: this region is in particular need of careful review.)

The SEC RR for a name SHOULD be supplied optionally in the additional data section if the CD bit is not set whenever a zone's NS or KEY set is requested. If a request for a KEY set is sent to a parent-only server and the server is not recursive, the server should add the SEC record to the additional section of the referral message.

If a name server authoritative for a child zone is asked for its SEC RR and the server has never learned the SEC RR (whether through caching the record or by also loading the parent zone), the server MAY answer with a negative answer. The resolver seeking a SEC RR SHOULD know to ask for this from a parent-serving name server.

2.1 Negative Answer Bitmap

The Negative Answer Bitmap indicates the mechanism for use in denying the existance of data. The bitmap is 16 bits, the most significant bit called 0, least significant is 15.

```
Bit  0 = The parent doesn't know what the child uses (1=Yes)
Bit  1 = The child signs its negative answers (1=Yes)
Bit  2 = The child follows traditional DNS rules (1=yes)
```

```
   Bit  3 = The child uses the NXT record (1=yes)
   Bit 14 = The child uses a locally defined mechanism (1=yes)
   Bit 15 = The length of the bit field has been extended (1=yes)
```

Bits 4 through 14 are currently unassigned, and are under the purview of IANA. Bit 15 MUST BE zero. (This specification must be superceeded to define an extension mechanism.)

A zone may use multiple mechanisms to indicate a negative answer. A zone SHOULD expect that a resolver finding any one of the mechanisms used in a reply indicates a negative answer, i.e. the mechanisms are OR'd together.

The only illegal values for this bit field are:

```
   Bit 0 = 1 and any other bit turned on
   Bit 0 = 0, Bit 1 = 1, and no other bit turned on
   Bit 15 = 1
```

2.1.1 Bit 0 (Better titles will be attached later)

The situation in which this bit is on should not arise, but it is defined to be safe. The philosophy behind this is that security

parameters should always be made explicit, including when a sitation is unclear.

2.1.2 Bit 1

This bit indicates that the child attachs SIG records to the resource records used in the negative answer. For example, this may indicate that the reslover should expect to see a SIG (NXT) when an NXT is returned.

2.1.3 Bit 2

The child will answer with an SOA or any of the other means used in the past to indicate a negative answer. (I think a reference to the negative answer/cache document should go here.)

2.1.4. Bit 3

The child uses the NXT as defined in RFC 2535. This document declares that the use of the NXT is optional, a deviation from RFC 2535.

2.1.5 Bit 14

The child is using a mechanism that is not globally defined. A zone should be in such a state for only experimental reasons and realize that in this state, the negative answers it gives may not be useful to

the general population of resolvers.

2.1.6 Bit 15

As of this specification, this bit must be 0 (zero).

2.1.7 Unallocated bits

The remainder of the bits must also be zero. A procedure will be defined for allocating them.

2.2 Security Parameters

The Security parameters is a sequence of options and values. An option is a numeric indicatior of the parameter. The value is usually either a yes or no, or a enumerated value. In rare instances, an option may require variable length data, in this case a triplet of option-length-value is used. The presence of the length field is indicated by the most significant bit in the option field being 1. Due to the nature of the SEC RR, the length field is not commonly used.

The option field is 8 bits. The most significant bit of the options field is turned on if there is a length field. The value field is also 8 bits. If the option-length-value is needed, the length is 8 bits and contains the number of octets comprising the value. No padding is used.

An option may appear multiple times in the Security Parameters. The sequencing of the options is not significant. If two options

contradict each other, this is an error, and is noted by the resolver. A self-contradictory SEC RR is a security error and data from the zone covered by it SHOULD be considered at risk.

Option Values are

0	Reserved
1	Zone is unsigned
2	Key Algorithm in use
3	Signing policy
0x70-0x7F	Locally defined (no length field)
0xF0-0xFF	Locally defined (uses length field)

All unassigned option values are under the control of IANA. Values 0 to 127 do not use the length field, values 128 to 255 do use the length field. The option value is to be treated as unsigned.

2.2.1 Option 0

This option is reserved for future definition.

2.2.2 Option 1

The parent has not signed a KEY RR for the child, therefore the child zone has no DNSSEC approved signing keys. If this option is not present, then the resolver SHOULD assume that there are zone keys in the child zone.

If the value of this is non-zero, this assertion is true. If the value is zero, this assertion is false. If the parent has signed keys for the child, the value is zero, however, in this case, the parent SHOULD NOT include this option in the security parameters.

It is tempting to exclude an unsigned zone option from this list, relying on the absence of any in use key algorithms (option 2) to imply that the zone is unsigned. The unsigned option is included to make this information explicit, so that when analyzing a running zone, it is obvious to an administrator that a zone is unsigned.

2.2.3 Option 2

The parent has signed a key for the child which claims a particular algorithm. This value field is equal to that of the algorithm field of the triggering KEY RR.

Option 2 can be repeated for different algorithms. It is not necessary to have multiple Option 2 entries with the same key algorithm value.

If Option 1 and Option 2 appear in the same SEC RR, this is a self-contradictory record. If neither Option 1 nor Option 2 appear, this also constitues a self-contradictory record.

2.2.4 Option 3

The child has the option to require that all material signatures (those generated by DNSSEC-approved signing keys) must be validated (within any temporal constraints) for the data to be considered valid. The child may instead require that just one of the signatures be validated. This may be a reflection of the manner in which a zone's administration is shared amongst organizations.

If Option 3 is not present (and Option 2 is), the resolver SHOULD assume that ALL (temporally valid) signatures are required. If the parent includes at least one Option 2, it SHOULD specify an Option 3, with a value indicated by the child.

Values for Option 3 are

0	Reserved
1	All signatures are required
2	One signature is required
256	Locally defined

All remaining values are under the control of IANA.

(Editorial note: whether the assumption that all signatures are necessary or just one is sufficient in the absence of this option is open to WG debate.)

2.2.5 Options 0x70-0x7F

This option is reserved for an organization to use locally, in an experimental fashion. This option does not use the length field. Global interpretation of this option is undefined.

2.2.6 Options 0xF0-0xFF

This option is reserved for an organization to use locally, in an experimental fashion. This option uses the length field. Global interpretation of this option is undefined.

3. Master File Representation

The SEC RR fields are to be represented as hexidecimal fields, with a preceeding '0x', or in decimal format. Hexidecimal SHOULD be used.

For example, the SEC RR representing a zone that use signed NXT records, and has one or more DSA keys, one or more RSA keys, and requires that just one signature be verified would be:

```
myzone.test.   3500 IN SEC 0x5000 0x0201 0203 0302
```

(0x020102030302 is one field, hence one 0x prefix.)

Hex values for the security parameters MAY BE separated by whitespace, as shown. DNS data display routines SHOULD substitute

mnemonics for these values, but MUST write the numeric form in master files.

4. Signature Policy

The SEC RR must be signed by one or more zone keys of the parent (delgating) zone, and the signatures must adhere to the parent's policy.

The SEC RR for the root zone is the lone exception, it appears at the apex of the root zone, and must be signed sufficiently by the root's zone key or keys.

5. Cache Considerations

When a SEC RR set for a name is held in a cache, it will have a credibility rating indicating that the data came from the parent (unless the parent and child share servers). When data about the same name arrives from the child, with a higher credibility, the newly arrived data MUST NOT cause the cache to remove the SEC RR.

6. IANA Considerations

IANA is requested to assign this RR an type parameter for DNS, and to assign the indicated option numbers and values when requests are approved. The procedure for requesting new options and values will be defined in future versions of this specfication.

7. Security Considerations

This record is designed to address the concerns of securing delegation points and resolving the security of DNS answers. This record is important to the security because it supplies needed information and eases the burden of security on the DNS.

The SEC RR does require one piece of additional information not addressed to date to be communicated from the parent to the child. This is the signature policy. This will be needed in the operations documents.

Editorial Note: This document would benefit by a companion document describing the process of evaluating the signatures in DNS. Such a document would provide clearer input to the security parameters field.

8. Editorial Considerations

Although somewhat detailed in this current description, this record is still in the formative state. The -00 document has been quickly written to test the waters for interest.

9. References

RFC 2535 is the prime DNSSEC definition. RFC 2181 is the Clarify document. EDNS0 reference needed...

10. Acknowledgements

This record is a successor to the Zone Key Referral, originally promoted by John Gilmore and Jerry Scharf. A DNSSEC workshop sponsored by the NIC-SE in May 1999 provided the enlightenment that expanded the Zone Key Referral into the SEC RR proposal.

11. Author's Addresses

Edward Lewis	Jerry Scharf	Olafur Gudmundsson
NAI Labs	Internet Software Consortium	NAI Labs
3060 Washington Road	950 Charter St	3060 Washington Rd
Glenwood, MD 21738	Redwood City, CA 94063	Glenwood, MD 21738
+1 443 259 2352	+1 650 779 7060	+1 443 259 2389
<lewis@tislabs.com>	<scharf@vix.com>	<ogud@tislabs.com>

12. Full Copyright Statement