INTERNET-DRAFT Donald E. Eastlake 3rd
UPDATES RFC 2535 IBM

Expires: June 2000 December 1999
draft-ietf-dnsind-sig-zero-01.txt

DNS Request and Transaction Signatures (SIG(0)s)

--- ------- --- ----------- ---------- - ------- -

Status of This Document

This draft, file name draft-ietf-dnsind-sig-zero-01.txt, is intended to become a Proposed Standard RFC updating Proposed Standard [RFC 2535]. Distribution of this document is unlimited. Comments should be sent to the DNS Working Group mailing list <namedroppers@internic.net> or to the author.

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. 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.''

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.

Abstract

Extensions to the Domain Name System (DNS) are described in [RFC 2535] that can provide data origin and transaction integrity and authentication to security aware resolvers and applications through the use of cryptographic digital signatures.

Implementation experience has indicated the need for minor but non-interoperable changes in Request and Transaction signature resource records (SIG(0)s). These changes are documented herein.

Acknowledgments

The significant contributions and suggestions of the following persons (in alphabetic order) to this draft are gratefully acknowledged:

Olafur Gudmundsson
Brian Wellington

Table of Contents

1. Introduction

This document makes minor but non-interoperable changes to part of [RFC 2535], familiarity with which is assumed, and includes additional explanatory text. These changes concern SIG Resource Records (RRs) that are used to sign DNS requests and transactions / responses. Such a resource record, because it has a type covered field of zero, is frequently called a SIG(0). The changes are based on implementation and attempted implementation experience with TSIG [draft-ietf-dnsind-tsig-*.txt] and the [RFC 2535] specification for SIG(0).

Sections of [RFC 2535] updated are all of 4.1.8.1 and parts of 4.2 and 4.3. No changes are made herein related to the KEY or NXT RRs or to the processing involved with data origin and denial authentication for DNS data.

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 [RFC 2119].

2. SIG(0) Design Rationale

The authenticated data origin and data existence denial services of secure DNS protect only data resource records (RRs) or authenticatably deny their nonexistence. These services provide no protection for DNS requests, no protection for message headers on requests or responses, and no protection of the overall integrity of a response.

If header bits are falsely set by a bad server, there is little that can be done. However, it is possible to add transaction and query authentication to be sure that queries and responses and not tampered with in transit.

2.1 Transaction Authentication

Transaction authentication means that a requester can be sure it is at least getting the messages from the server it queried and that the response is from the request it sent (i.e., that these messages have not been diddled in transit). This is accomplished by optionally adding either a TSIG RR [draft-ietf-dnsind-tsig-*.txt] or, as described herein, a SIG(0) resource record at the end of the response which digitally signs the concatenation of the server's response and the corresponding resolver query.

2.2 Query Authentication

Requests can also be authenticated by including a TSIG or, as described herein, a special SIG(0) RR at the end of the request. Authenticating requests serves no function in DNS servers the predate the specification of dynamic update. Requests with a non-empty additional information section produce error returns or may even be ignored by a few such older DNS servers. However, this syntax for signing requests is defined for authenticating dynamic update requests [RFC 2136], TKEY requests [draft-ietf-dnsind-tkey-*.txt], or future requests requiring authentication.

2.3 Keying

The private keys used in transaction security belong to the host composing the DNS response message, not to the zone involved. Request authentication may also involve the private key of the host or other entity composing the request or of a zone to be affected by the request or other private keys depending on the request authority it is sought to establish. The corresponding public key(s) are normally stored in and retrieved from the DNS for verification as KEY RRs with a protocol byte of 3 (DNSSEC) or 255 (ANY).

Because requests and replies are highly variable, message authentication SIGs can not be pre-calculated. Thus it will be necessary to keep the private key on-line, for example in software or in a directly connected piece of hardware.

2.4 Differences Between TSIG and SIG(0)

There are significant differences between TSIG and SIG(0).

Because TSIG involves secret keys installed at both the requester and server the presence of such a key implies that the other party understands TSIG and likely has the same key installed. Furthermore, TSIG uses keyed hash authentication codes which are relatively inexpensive to compute. Thus it is common to authenticate requests with TSIG and responses are authenticated with TSIG if the corresponding request is authenticated.

SIG(0) on the other hand, uses public key authentication, where the public keys are stored in DNS as KEY RRs. Thus, existence of such a KEY RR does not necessarily imply implementation of SIG(0). In addition, SIG(0) involves relatively expensive public key cryptographic operations that should be minimized and the

verification of a SIG(0) involves obtaining and verifying the

corresponding KEY which can be an expensive and lengthy operation. Indeed, a policy of using SIG(0) on all requests and verifying it before responding would, for some configurations, lead to a deadly embrace with the attempt to obtain and verify the KEY needed to authenticate the request SIG(0) resulting in additional requests accompanied by a SIG(0) leading to further requests accompanied by a SIG(0), etc. Furthermore, omitting SIG(0)s when not required on requests halves the number of public key operations required by the transaction.

For these reasons, SIG(0)s SHOULD only be used on requests when necessary to authenticate that the requester has some required privilege or identity. SIG(0)s on replies are defined in such a way as to not require a SIG(0) on the corresponding request and still provide transaction protection. Some replies, such as those involving TKEY [draft-ietf-dnsind-tkey-*.txt], MUST be authenticated with TSIG or SIG(0). For other replies, whether they are authenticated by the server or required to be authenticated by the requester SHOULD be a local configuration option.

3. The SIG(0) Resource Record

The structure of and type number of SIG resource records (RRs) is given in [RFC 2535] Section 4.1. However all of Section 4.1.8.1 and the parts of Sections 4.2 and 4.3 related to SIG(0) should be considered replaced by the material below. Any conflict between [RFC 2535] and this document concerning SIG(0) RRs should be resolved in favor of this document.

For all transaction SIG(0)s, the signer field MUST be the name of the originating server host and there MUST be a KEY RR at that name with the public key corresponding to the private key used to calculate the signature. (The inverse IP address mapping name MAY be used if the relevant KEY is stored there.)

For all SIG(0) RRs, the owner name, class, TTL, and original TTL, are meaningless. The TTL fields SHOULD be zero and the CLASS filed SHOULD be ANY. To conserve space, the owner name SHOULD be root (a single zero octet). When SIG(0) authentication on a response is desired, that SIG RR must be considered the highest priority of any additional information for inclusion in the response. If the SIG(0) RR cannot be added without causing the message to be truncated, the server MUST alter the response so that a SIG(0) can be included. This response consists of only the question and a SIG(0) record, and has the TC bit set and RCODE 0 (NOERROR). The client SHOULD at this point retry the request using TCP.

3.1 Calculating Request and Transaction SIGs

A DNS request may be optionally signed by including one or more SIG(0)s at the end of the query additional information section. Such SIGs are identified by having a "type covered" field of zero. They sign the preceding DNS request message including DNS header but not including the UDP/IP header or any request SIG(0)s at the end and before the request RR counts have been adjusted for the inclusions of any request SIG(0)s.

Note: requests and response can either have a TSIG or one or more SIG(0)s but not both a TSIG and a SIG(0).

They are calculated by using a "data" (see [RFC 2535], Section 4.1.8) of (1) the SIG's RDATA section omitting the signature subfield itself, (2) the entire DNS query messages, including DNS header, but not the UDP/IP header or any SIG(0) and before the reply RR counts have been adjusted for the inclusion of any SIG(0). That is

```
      data = RDATA | request - SIG(0)s
```

where "|" is concatenation and RDATA is the RDATA of the SIG(0) being calculated less the signature itself.

Similarly, a SIG(0) can be used to secure a response and the request that produced it. Such transaction signatures are calculated by using a "data" of (1) the SIG's RDATA section omitting the signature itself, (2) the entire DNS query message that produced this response, including the query's DNS header and any SIG(0)s but not its UDP/IP header, and (3) the entire DNS response message, including DNS header but not the UDP/IP header or any SIG(0) and before the response RR counts have been adjusted for the inclusion of any SIG(0).

That is

```
data = RDATA | full query | response - SIG(0)s
```

where "|" is concatenation and RDATA is the RDATA of the SIG(0) being calculated less the signature itself.

Verification of a response SIG(0) (which is signed by the server host key, not the zone key) by the requesting resolver shows that the query and response were not tampered with in transit, that the response corresponds to the intended query, and that the response comes from the queried server.

In the case of a DNS message via TCP, a SIG(0) on the first data packet is calculated with "data" as above and for each subsequent packet, it is calculated as follows:

```
data = RDATA | DNS payload - SIG(0)s | previous packet
```

where "|" is concatenations, RDATA is as above, and previous packet is the previous DNS payload including DNS header and any SIG(0)s but not the TCP/IP header. Support of SIG(0) for TCP is OPTIONAL. As an alternative, TSIG may be used after, if necessary, setting up a key with TKEY [draft-ietf-dnsind-tkey-*.txt].

Except where needed to authenticate an update, TKEY, or similar privileged request, servers are not required to check request SIGs.

3.2 Processing Responses and SIG(0) RRs

If a SIG RR is at the end of the additional information section of a response and has a type covered of zero, it is a transaction signature covering the response and the query that produced the response. For TKEY responses, it MUST be checked and the message rejected if the checks fail. For all other responses, it MAY be

checked and the message rejected if the checks fail.

If a response SIG(0) checks succeed, such a transaction authentication SIG does NOT directly authenticate the validity any data-RRs in the message. However, it authenticates that they were sent by the queried server and have not been diddled. (Only a proper SIG(0) RR signed by the zone or a key tracing its authority to the zone or to static resolver configuration can directly authenticate data-RRs, depending on resolver policy.) If a resolver or server does not implement transaction and/or request SIGs, it MUST ignore them without error where they are optional and treat them as failing where they are required.

3.3 SIG(0) Lifetime and Expiration

The inception and expiration times in SIG(0)s are for the purpose of resisting replay attacks. They should be set to form a time bracket such that messages outside that bracket can be ignored. In IP networks, this time bracket should not normally extend further than 5 minutes into the past and 5 minutes into the future.

4. Security Considerations

No additional considerations beyond those in [RFC 2535].

The inclusion of the SIG(0) inception and expiration time under the signature improves resistance to replay attacks.

5. IANA Considerations

No new fields are created or field values assigned by the document.

References

[RFC 1982] - Robert Elz, Randy Bush, "Serial Number Arithmetic", 09/03/1996.

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

[RFC 2136] - P. Vixie, S. Thomson, Y. Rekhter, J. Bound, "Dynamic Updates in the Domain Name System (DNS UPDATE)", 04/21/1997.

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

[draft-ietf-dnsind-tsig-*.txt] - P. Vixie, O. Gudmundsson, D. Eastlake, B. Wellington, "Secret Key Transaction Signatures for DNS (TSIG)".

[draft-ietf-dnsind-tkey-*.txt] - D. Eastlake, "Secret Key Establishment for DNS (RR)"

Author's Address

Donald E. Eastlake 3rd
IBM
65 Shindegan Hill Road
Carmel, NY 10512 USA

Telephone: +1-914-276-2668(h)
+1-914-784-7913(w)
fax: +1-914-276-2947(h)
email: dee3@us.ibm.com

Expiration and File Name

This draft expires June 2000.

Its file name is draft-ietf-dnsind-sig-zero-01.txt.

Appendix: SIG(0) Changes from RFC 2535

Add explanatory text concerning the differences between TSIG and SIG(0).

Change the data over which SIG(0) is calculated to include the SIG(0) RDATA other than the signature itself to secure the signature inception and expiration times and resist replay attacks. Specify SIG(0) for TCP.

Add discussion of appropriate inception and expiration times for SIG(0).

Change wording to permit mixing TSIG and SIG(0) RRs.

Reword some areas for clarity.