DNSOP
Internet-Draft
Obsoletes: 2931 (if approved)
Intended status: Informational
Expires: 25 January 2026

D. Eastlake
Independent
J. Stenstam
Swedish Internet Foundation

24 July 2025

Domain Name System (DNS) Public Key Based Request and Transaction Authentication (SIG(0))

draft-eastlake-dnsop-rfc2931bis-sigzero-03

Abstract

This document specifies use of the Domain Name System (DNS) SIG Resource Record (RR) to provide a public key based authentication of DNS requests and transactions. Such a resource record, because it has a "type covered" field of zero, is frequently called a "SIG(0)". This document obsoletes RFC 2931.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on 25 January 2026.

Table of Contents

1. Introduction

This document specifies use of the Domain Name System (DNS) SIG Resource Record (RR) to provide a public key based method to authenticate DNS requests and transactions. Such a resource record, because it has a "type covered" field of zero, is frequently called a "SIG(0)". This document obsoletes [RFC2931].

1.1. Terminology

General familiarity with DNS terminology [RFC9499] is assumed in this document.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. SIG(0) Design Rationale and Use

SIG(0) provides authentication and integrity protection for DNS transactions and requests that is not provided by the DNSSEC RRSIG RR specified in [RFC4034]. The authenticated data origin services of DNSSEC [RFC4035] either provide protected data resource records (RRs) or authenticatable denial of their existence. Those services provide no protection for

* glue records,
* DNS requests,
* the DNS message headers on requests or responses, and
* the overall integrity of a transaction, that is to say, that the response was issued in response to the request sent and neither was tampered with.

Transaction authentication provides some cryptographic assurance to a resolver that it is at least getting the DNS response message from the server and that the received message is in response to the request it sent. This is accomplished by adding either a TSIG RR [RFC8945] or, as specified herein, a SIG(0) RR, at the end of the additional information section of the response which authenticates the concatenation of the corresponding resolver request and the server’s response.

Requests can also be authenticated by including a SIG(0) RR at the end of the request’s additional information section. Authenticating requests served little purpose in DNS servers before the specification of dynamic update except to enable more rigorous enforcement of restrictions on which resolvers can make what requests of the server. The method of signing requests specified herein is primarily intended for authenticating dynamic update requests [RFC3007], TKEY requests [rfc2930bis], or other requests specified in the future that require authentication.

SIG(0) is extensively used in [RFC9665].

Depending on the request authority it is sought to establish, the private keys used in public key based request security belong to the host composing the DNS request message or other entity composing the request or to a zone to be affected by the request. The corresponding public key(s) can be stored in and retrieved from the DNS for verification as KEY RRs with a protocol byte of 3 or 255 (ANY).

3. Differences Between TSIG and SIG(0)

A TSIG [RFC8945] RR can also be used for request and transaction authentication. There are significant differences between TSIG and SIG(0).

Because TSIG involves secret keys available at both the requester and server the presence of such a key can imply 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 to authenticate responses with TSIG if the corresponding request is authenticated.

SIG(0) on the other hand, uses public key authentication, where the public keys can be stored in DNS as KEY RRs and a private key is stored at the signer. Existence of such a KEY RR does not necessarily imply that SIG(0) is implemented or enabled. 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 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 transactions are defined in such a way as to not require a SIG(0) on the corresponding request and still provide transaction protection. Which SIG(0)s are authenticated by the server or required to be authenticated by the requester SHOULD be a local configuration option.

A request or response may have either a TSIG or one or more SIG(0) RRs. It MUST NOT have both a TSIG and a SIG(0) and a request with both MUST be rejected with FORMERR. A response with both is ignored.

> | Both DNS transaction security and DNSSEC data security originally used the SIG (type = 24) and KEY (type = 25) RRs. DNSSEC was changed to use the RRSIG (type = 46) and DNSKEY (type = 48) RRs [RFC4034]; however, transaction security, including TKEY [rfc2930bis] and SIG(0), continue to use the SIG and KEY RRs.

4. The SIG(0) Resource Record

The structure of SIG resource records (RRs) is the same as the structure of the RRSIG RR in [RFC4034] Section 4 except as provided below.

The type of the SIG RR is 24. For SIG(0) RRs, the owner name, class, TTL, and original TTL, and Labels fields are meaningless. The TTL fields and Labels field MUST be zero, the CLASS field MUST be 255 (ANY), and these fields are ignored on receipt. A TTL of zero decreases the risk that a DNS implementation that does not understand SIG(0) will cache such an RR. To conserve space, the owner name SHOULD be root (a single zero octet).

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

For all transaction SIG(0)s, there MUST be a KEY RR associated with the signer name with the public key corresponding to the private key used to calculate the signature. For general transaction authentication and integrity, the signer field is a name of the originating host; for example, the host domain name used may be the inverse IP address mapping name for an IP address of the host if the relevant KEY is stored there. For SIG(0)s on requests requiring authentication, the signer relates to the authority being requested such as authority to update a zone.

When SIG(0) authentication on a response is desired, the SIG(0) 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, set the TC bit, and return RCODE 0 (NOERROR). The client should at this point retry the request using an available transport that does not have UDP packet size restrictions.

4.1. EDNS Option Structure

SIG(0) does not provide for an Original ID field as provided in TSIG [RFC8945], which is needed to validate the SIG(0) on a forwarded request (see Section 6.1), and does not provide an error field as provided in the TSIG and TKEY [rfc2930bis] RRs. These can be carried in an EDNS(0) RR option [RFC6891] with option code TBD.

This EDNS(0) option may occur multiple times if there are multiple SIG(0)s.

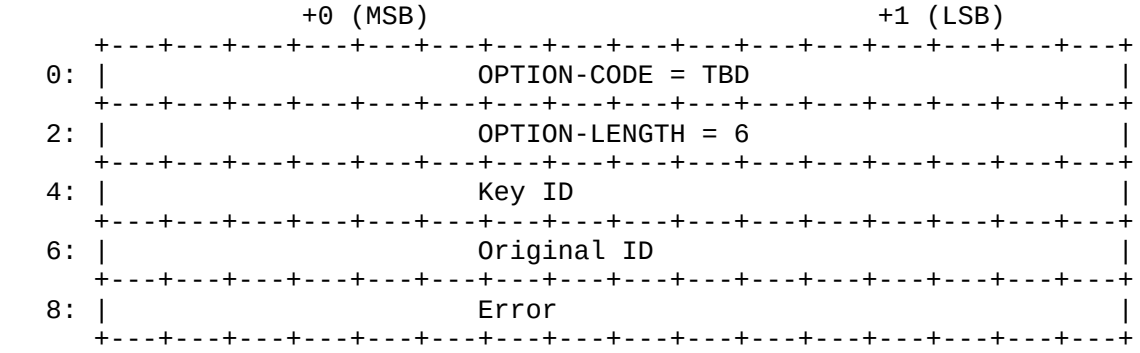

Figure 1: SIG(0) Original ID and Error Return EDNS Option

The Key ID field is used to tie each EDNS option to the corresponding SIG(0) RR. KeyIds are not guaranteed to be unique, so it is the responsibility of the SIG(0) creator to not use multiple keys with the same KeyId.

The Error field, in responses, is an unsigned 16-bit integer containing the extended RCODE covering SIG(0) processing. In requests, this MUST be zero.

5. Calculating Request and Transaction SIG(0)s

A DNS request may be optionally signed by including one or more SIG(0) RRs at the end of the request additional information section. They are calculated for and sign the "data" consisting of the following:

1. the SIG(0)'s RDATA section omitting the signature subfield itself, not just zeroing its value, and
2. the DNS request message, including DNS header, but not the UDP/IP header and before any SIG(0) RR has been added so that, for example, counts have not yet been adjusted for SIG(0) inclusion.

That is

```
data = (SIG(0) RDATA) | (request without SIG(0)s)
```

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

Similarly, a SIG(0) can be used to secure a response and the request that produced it. Such a transaction signature is calculated by using a "data" consisting of

1. the SIG(0)'s RDATA section omitting (not just zeroing) the signature itself,
2. the entire DNS request message that produced this response, including the request's DNS header and any SIG(0)s that were present but not its UDP/IP header, and
3. the entire DNS response message, including DNS header but not the UDP/IP header and before any SIG(0)s have been added so that, for example, response RR counts have not yet been adjusted for SIG(0) inclusion.

```
data =
  (SIG(0) RDATA) | full request | (response without SIG(0))
```

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

Successful verification of a response SIG(0) 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 a stream transport (e.g. TCP, DoT, etc.), 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 = (SIG(0) RDATA) | (DNS payload without SIG(0)) | previous packet
```

where "|" is concatenations, RDATA is as above, and previous packet is the previous DNS payload including DNS header and SIG(0) but not the TCP/IP header.

Except where needed to authenticate an update, TKEY, or similar privileged request, servers are not required to check a request SIG(0).

6. Processing SIG(0) RRs and Responses

If one or more SIG RRs are at the end of the additional information section of a response and have a type covered of zero, they are transaction signatures covering the response and the request that produced the response.

If the time when a SIG(0) on a request is received is outside the interval indicated by the inception and expiration times in the SIG(0), the BADTIME error is returned.

If a response's SIG(0) check succeeds, such a transaction authentication SIG does NOT directly authenticate the validity of any data RRs in the message. However, it authenticates that they were sent by the queried server and have not been altered in transit. (Only an RRSIG RR [RFC4034] signed by the zone or a key tracing its authority to the zone or to resolver configuration can directly authenticate data RRs, depending on resolver policy.) If a resolver or server does not implement transaction and/or request SIG(0)s, it MUST ignore them without error where they are optional and treat them as failing where they are required.

6.1. Special Considerations for Forwarding Servers

A server acting as a forwarding server of a DNS message SHOULD check for the existence of a SIG(0) record. If it cannot verify the SIG(0), the server MUST forward the message unchanged including the SIG(0) and the corresponding EDNS(0) option if one is present. If the SIG(0) passes all checks and verifies, the forwarding server MUST, if possible, add a SIG(0) of its own to the destination or the next forwarder. If no transaction security is available to the destination and the message is a request, and if the corresponding response has the AD flag (see [RFC4035]) set, the forwarder MUST clear the AD flag before adding the SIG(0) to the response and returning the result to the system from which it received the request.

A forwarded SIG(0) is not verifiable unless the original transaction ID is preserved by, for example,

* using TCP and maintaining a separate ID space for that TCP connection between the forwarder and the server or next forwarder or

* by forwarding the EDNS option specified in Section 4.1.

7. Security Considerations

Private keys used to create SIG(0) RRs are very sensitive information and all available steps should be taken to protect them on every host on which they are stored. Such hosts may need to be physically protected. If they are multi-user machines, great care should be taken so that unprivileged users have no access to private keying material.

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

8. IANA Considerations

IANA is requested to assign an EDNS OPT number in the "DNS EDNS0 Option Codes (OPT)" Registry as follows:

Value	Name	Status	Reference
TBD	SIGZERO	Standard	[this document]

Table 1

9. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Resource Records for the DNS Security Extensions", RFC 4034, DOI 10.17487/RFC4034, March 2005, <https://www.rfc-editor.org/info/rfc4034>.

[RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Protocol Modifications for the DNS Security Extensions", RFC 4035, DOI 10.17487/RFC4035, March 2005, <https://www.rfc-editor.org/info/rfc4035>.

[RFC6891] Damas, J., Graff, M., and P. Vixie, "Extension Mechanisms for DNS (EDNS(0))", STD 75, RFC 6891, DOI 10.17487/RFC6891, April 2013, <https://www.rfc-editor.org/info/rfc6891>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

10. Informative References

[rfc2930bis] Eastlake, D. and M. Andrews, "Secret Key Agreement for DNS (TKEY Resource Record)", work in process, <https://datatracker.ietf.org/doc/draft-eastlake-dnsop-rfc2930bis-tkey/>.

[RFC2931] Eastlake 3rd, D., "DNS Request and Transaction Signatures (SIG(0)s)", RFC 2931, DOI 10.17487/RFC2931, September 2000, <https://www.rfc-editor.org/info/rfc2931>.

[RFC3007] Wellington, B., "Secure Domain Name System (DNS) Dynamic Update", RFC 3007, DOI 10.17487/RFC3007, November 2000, <https://www.rfc-editor.org/info/rfc3007>.

[RFC8945] Dupont, F., Morris, S., Vixie, P., Eastlake 3rd, D., Gudmundsson, O., and B. Wellington, "Secret Key Transaction Authentication for DNS (TSIG)", STD 93, RFC 8945, DOI 10.17487/RFC8945, November 2020, <https://www.rfc-editor.org/info/rfc8945>.

[RFC9499] Hoffman, P. and K. Fujiwara, "DNS Terminology", BCP 219, RFC 9499, DOI 10.17487/RFC9499, March 2024, <https://www.rfc-editor.org/info/rfc9499>.

[RFC9665] Lemon, T. and S. Cheshire, "Service Registration Protocol for DNS-Based Service Discovery", RFC 9665, DOI 10.17487/RFC9665, June 2025, <https://www.rfc-editor.org/info/rfc9665>.

Appendix A. Changes from [RFC2931]

1. Add section on considerations for forwarding servers.

2. Remove statement that TCP support for SIG(0) is OPTIONAL.
3. Allow multiple SIG(0)s in a DNS request/response.
4. Specify an EDNS option to convey the original ID and return an extended error code.
5. Editorial changes including updates to meet current Internet draft format requirements. Update references. Convert source to XMLv3.

Appendix B. Change History

RFC Editor: Please delete this section before publication.

B.1. From [RFC2931] to -00

1. Change to require KEY RRs used in connection with SIG(0) to have a protocol byte of 255 (ANY). ([RFC2931] also permits a protocol byte of 3.
2. Change implementation requirement for the TTL and CLASS field of SIG(0) RRs from SHOULD be zero and 255, respectively, to MUST have those values and are ignored on receipt.
3. Add section on considerations for forwarding servers.
4. Remove statement that TCP support for SIG(0) is OPTIONAL.
5. Specify an EDNS option to convey the original ID and return an extended error code.
6. Editorial changes including updates to meet current Internet draft format requirements. Update references. Convert source to XMLv3.

B.2. From -00 to -01

1. Add section on error return via EDNS and add IANA request for an EDNS OPT number.
2. Clarify that a SIG(0) public key can be associated with a zone or otherwise indicate authorization.
3. Add author.
4. Editorial Changes.

B.3. From -01 to -02

1. Permit multiple SIG(0)s.
2. Back out change requiring protocol 255 in SIG(0)s and again permit protocol 3 or 255.
3. Add reference to SIG(0) usage in SRP.
4. Editorial Changes.

B.4. From -02 to -03

1. Generalize TCP references to include mentions of other stream protocols.

2. Update reference to DNSSD SRP from draft to [RFC9665].

3. Editorial Changes.

Acknowledgements

The comments and suggestions of the following are gratefully acknowledged:

tbd

The comments and suggestions of the following persons were incorporated into [RFC2931], which was the previous version of this document, and are gratefully acknowledged:

Olafur Gudmundsson, Ed Lewis, Erik Nordmark, Brian Wellington.

Authors' Addresses

Donald E. Eastlake 3rd
Independent
2386 Panoramic Circle
Apopka, Florida 32703
United States of America
Phone: +1-508-333-2270
Email: d3e3e3@gmail.com

Johan Stenstam
Swedish Internet Foundation
Email: johan.stenstam@internetstiftelsen.se