DNS SIG(0)

DNS Public Key Based Request and Transaction Authentication

draft-eastlake-dnsop-rfc2931bis-sigzero-01

Donald Eastlake 3rd d3e3e3@gmail.com

Johan Stenstam
johan.stenstam@internetstiftelsen.se

TSIG and SIG(0)

- TSIG (Transaction Signature, RFC 8945) is a meta RR providing efficient DNS request and transaction authentication based on a keyed hash algorithm and shared secret key. (transaction = concatenated request and response)
- When no shared secret is in place, SIG(0) provides the same sort of services using public/private key signatures.

SIG(0) RFC 2931

- Uses the SIG RR (which has the structure of the RRSIG RR) with a “Type covered” field of zero.
- SIG(0) signs requests and transactions using a private key for which the authenticator has the public key.
- Like TSIG [RFC 8945] but can be used when there is no shared secret in place.
  - Can authenticate general requests and replies if the public key is associated with requester/server host.
  - Can authorize UPDATE or the like if public key associated with zone or other authority.
  - Public keys may be stored in the DNS with the KEY RR.

Problems with SIG(0)

- Has no Error field.
- Has no Original ID field so forwarded authentication not obviously supported.
  - BIND supports multi-hop SIG(0) by forwarding with TCP with the same ID and maintaining a separate ID space per TCP connection.

Changes in rfc2931bis SIG(0)

- Removed statement that TCP support of SIG(0) is optional.
- Changed some implementation requirements to reduce the variability in SIG(0) RRs.
- Added section on considerations for forwarding servers.
- Added an EDNS(0) option to carry the Original ID and possibly return an error value.

Questions

- Questions for rfc2931bis, SIG(0)
  1. Handling Original ID and Error value return options:
     A. Use an EDNS(0) option as in current -01 draft.
     B. Reuse TTL to hold Original ID and Error value. (Slight increase in chance of erroneously being cached.)
     C. Specify a new RR to handle these fields. (Requires ability to create and parse a new structure.)
  2. Should more than one SIG(0) be allowed?
     - Easy to specify so that each signs as if any others were absent.
     - Useful in some cases of forwarding.
     - No intent to change TSIG restrictions.

Questions for rfc2931bis

	Single SIG(0) allowed	Multiple SIG(0)s allowed
Use EDNS(0)	OK	Complex – must bind EDNS(0) option to SIG(0)
Use redefined TTL	OK	OK
Use a new RR	OK	OK

Next Steps

- Update draft depending on the answer to the questions above and WG feedback.
  - The authors are leaning towards allowing more than one SIG(0) which may be easier using a redefined TTL or new RR.

- Request Working Group Adoption

END

Donald Eastlake 3rd d3e3e3@gmail.com

Johan Stenstam
johan.stenstam@internetstiftelsen.se

Two Types of DNS Security

- DNSSEC Data Security provides authentication of data RRs or authenticated denial of their existence cryptographically linked to the zone owner.
- DNS Transaction Security provides authentication of DNS requests and DNS transactions (concatenation of request and response) cryptographically linked to the resolver and server or to the authority being invoked by the request. Uses TSIG or SIG(0)

Historical RR Type Note

- Both DNS transaction security and DNSSEC data security originally used the
  - SIG (type = 24) and
  - KEY (type = 25) RRs [RFC2535].
- DNSSEC was changed to use the
  - RRSIG (type = 46) and
  - DNSKEY (type = 48) RRs [RFC4034].
- The corresponding RRs have the same field structure as each other. Transaction security continues to use the SIG and KEY RRs.