Abstract

It is possible to force a DNS server to fragment its response such that a fragmentation reassembly attack can insert records into the response. This document uses TSIG with a well known key to defeat such attacks.

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 January 3, 2020.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.
1. Introduction

It is possible to force a DNS server to fragment its response such that a fragmentation reassembly attack can insert records into the response. This document uses TSIG with a well known key to defeat such attacks.

Using TSIG [RFC2845] with a well known key effectively adds a cryptographically strong checksum to every DNS message. When combined with DNS COOKIE [RFC7873] in the request, it is effectively impossible to guess the correct checksum for the response. When DNS COOKIE is not used, a 64 bit nonce can be added to the Other Data section of the request's TSIG to achieve the same protection as it is part of the data that is hashed.

Existing servers, that support TSIG, can be configured with the well known key allowing them to answer clients that send requests with the well known key.

Existing clients, including recursive servers, that support TSIG, can be configured to send queries with the well known key for servers they regularly talk to after testing to see if the server has been configured with the well known key.

Tools, like those used to generate the Alexa 1 Million - TSIG Well Know Key Handling Report (experimental) report referenced below, can be used periodically to generate lists of servers that support the well known TSIG key. Alternatively one can configure a server to
send the well known key and have a exclusion list. This would be a late stage solution.

Signaling should be added to DHCP and RA to identify which servers support the well known TSIG key.

This does not protect against attackers that can see the DNS requests.

2. The Well Known Key

The well known key has a owner name of "." and uses HMAC-SHA256 [RFC4635] as its algorithm with a key of 256 zero bits.

Should it become necessary to roll the well known key’s algorithm, an updated RFC should be published with new algorithm and matching key definition. The standard TSIG error response of NOTAUTH/BADALG can be used to signal to try alternate algorithms. It is possible for servers to support multiple algorithms simultaneously by trying each in turn. Not all existing servers support trying multiple algorithms/keys for the same name.

3. Using The Well Known Key

Clients should opportunistically attempt to use the well known key and if they get an expected error they should fallback to not using the well known key unless they have already had a successful transaction using the well known key or have a priori knowledge that the well known key is supported.

All members of a anycast cluster of servers should use the same well known keys.

When sending a request a 64 bit nonce should be added to the TSIG Other Data section to increase the entropy of the request. The TSIG Other Data section currently unused in TSIG requests.

4. Old Servers

STD 13 [RFC1034] [RFC1035] compliant servers that do not support TSIG will return FORMERR or ignore the TSIG in the request.

Old servers that support TSIG are expected to return NOTAUTH/BADKEY.

In practice other results are seen. The following contains the results of sending a plain DNS query and the above well known key to all DNS servers for the Alexa Top 1 Million sorted pairwise. It is regenerated monthly.
5. Security Considerations

Using TSIG as a cryptographic checksum to defeat UDP fragmentation attacks requires that there is sufficient entropy in the request to have enough different MACs generated. A plain DNS query only has the current time and DNS query id to provide entropy. Additional entropy is added by using DNS COOKIE and/or adding a nonce to the TSIG Other Data data of the request.

Using a well known TSIG key does not fully protect against an attacker that can see the request but it reduces the attack to a plain response race rather than one that allows preloading the fragmentation reassembly buffers.

6. References

6.1. Normative References


6.2. URIs

[1] https://ednscomp.isc.org/compliance/alexa1m-tsig-wkk.txt
Appendix A. Configuring Servers to support the Well Known TSIG Key

A.1. BIND 9

Add the following to named.conf. Some end-of-life versions do not support HMAC-SHA256.

```plaintext
key "." {
   algorithm hmac-sha256;
   secret "AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA=";
};
```

A.2. Other Vendors

Other DNS vendors please send me instructions for your servers.

Appendix B. Configuring Recursive Servers to send the Well Known TSIG Key

B.1. BIND 9

Define the key as above and add multiple of the following to named.conf for the servers you wish to send the well known TSIG key to.

```plaintext
server <prefix> { keys "."; }
```

B.2. Other Vendors

Other DNS vendors please send me instructions for your servers.

Author’s Address

M. Andrews
Internet Systems Consortium
950 Charter Street
Redwood City, CA  94063
US

Email: marka@isc.org