Abstract

This document describes an additional Email Authentication Method compliant with RFC 7601. The method consists in looking up the sender's IP in a DNS whitelist.

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 http://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 September 20, 2018.

Copyright Notice

Copyright (c) 2018 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 (http://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

One of the many checks that mail servers carry out is to query DNS whitelists (DNSWL, [RFC5782]). The lookup is based on the connecting client’s IP address, so this check can occur very early in an SMTP transaction. The result can be used to counterweight policies that typically occur at early stages too, such as the Sender Policy Framework (SPF, the last paragraph of Appendix D.3 of [RFC7208] is illustrated in Appendix A). In addition, the result of a DNSWL lookup can also be used at later stages; for example, a delivery agent can use it to estimate the spamminess of an email message. The latter possibility needs a place to collect query results for downstream use, which is precisely what the Authentication-Results header field aims at providing.

Results often contain additional data, encoded according to DNSWL-specific criteria. The present method considers only whitelists --one of the major branches considered by [RFC5782]. In case of DNSxL, the boundary MTA (see [RFC5598]) which carries out the check and possibly stores the result, has to be able to discern at least the color of "x", which is required to make accept/reject decisions.

Data conveyed in A and TXT records can be stored as result’s parameters. In effect, they are tantamount to local policies, albeit outsourced. Downstream agents need to know DNSWL-specific encoding to understand the meaning of that data. In order to smooth operations, this document endorses a usage of TXT fields consistent with other authentication methods.

2. Method Details

The following ptype.property items define the relevant parameters where additional data can be stored. They augment the "pass" result with information about the entry found.
dns.zone: DNSWL query root domain, which defines the meaning of the result. Note that an MTA can use a local mirror with a different name. The name stored here has to be the best available reference for all foreseeable downstream consumers.

policy.ip: The bit mask value received in type A response, in dotted quad. Multiple entries MAY be arranged in a comma-separated list.

policy.txt: The TXT record, if any. Multiple records are concatenated as usual. See Section 3 for the resulting content.

The result of the method states how the query did, up to the interpretation of the result. In particular, some DNSBLs are known to return special codes to signal over quota, for example 127.0.0.255. If the result producer cannot interpret that value, that case results in a false positive.

pass: The query successfully returned applicable records. The sender is whitelisted, up to differing interpretation.

none: The query worked but yielded no record, or returned NXDOMAIN, so the sender is not whitelisted.

temprerror: The DNS evaluation could not be completed due to some error that is likely transient in nature, such as a temporary DNS error, e.g., a DNS RCODE of 2, commonly known as SERVFAIL, or other error condition resulted. A later attempt may produce a final result.

permerror: The DNS evaluation cannot work because test entries don’t work, that is, DNSWL is broken, or because queries are overquota, e.g., a DNS RCODE of 5, commonly known as REFUSED, or a DNSWL-specific policy.ip was returned. A later attempt is unlikely to produce a final result. Human intervention is required.

3. TXT Record Contents

[RFC5782] mentions that TXT records describe the reason why IP addresses are listed in a DNSWL. Some kind of uniform format is necessary to make automated use of that data.

This document RECOMMENDS that DNSWLs put a domain name at the beginning of the text record, possibly followed by a white space and more text. The domain name would correspond to the DNS domain name.
used by or within the ADMD operating the relevant MTA, sometimes called the "organizational domain". If no domain name is known, DNSWLs MAY use a subdomain of .INVALID [RFC2606] where the leftmost label hints at why an address is whitelisted given that its operating organization is not known.

4. IANA Considerations

There is a registry of Email Authentication Methods created by RFC7601. The method described in this document is referred by Table 1, along with its ptype.property values.

<table>
<thead>
<tr>
<th>Method</th>
<th>ptype</th>
<th>property</th>
<th>Value</th>
<th>Status</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>dnswl</td>
<td>dns</td>
<td>zone</td>
<td>DNSWL publicly accessible query root domain</td>
<td>active</td>
<td>1</td>
</tr>
<tr>
<td>dnswl</td>
<td>policy</td>
<td>ip</td>
<td>type A response received (or comma-separated list thereof)</td>
<td>active</td>
<td>1</td>
</tr>
<tr>
<td>dnswl</td>
<td>policy</td>
<td>txt</td>
<td>type TXT query response</td>
<td>active</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 1: Email Authentication Method

A new ptype, "dns" is introduced in Table 2. It is meant to be used for properties related to the Domain Name System (DNS [RFC1034]), whose value cannot be exactly derived from the relevant authentication method specification.

<table>
<thead>
<tr>
<th>ptype</th>
<th>Definition</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dns</td>
<td>[this doc]</td>
<td>The property being reported belongs to the Domain Name System</td>
</tr>
</tbody>
</table>

Table 2: Email Authentication Property Type

This method reuses four of the values already defined in the Email Authentication Result Names associated registry. They are listed in Table 3.
### Table 3: Email Authentication Result Names

<table>
<thead>
<tr>
<th>Auth Method</th>
<th>Code</th>
<th>Specification</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>dnswl</td>
<td>pass</td>
<td>Sender is whitelisted, up to returned code interpretation</td>
<td>active</td>
</tr>
<tr>
<td>dnswl</td>
<td>none</td>
<td>NXDOMAIN or no record, sender is not whitelisted</td>
<td>active</td>
</tr>
<tr>
<td>dnswl</td>
<td>temperror</td>
<td>Transient DNS error during the query</td>
<td>active</td>
</tr>
<tr>
<td>dnswl</td>
<td>permerror</td>
<td>Query cannot work, human intervention needed</td>
<td>active</td>
</tr>
</tbody>
</table>

5. Security Considerations

All of the considerations described in Section 7 of [RFC7601] apply.

In addition, the usual caveats apply about importing text from external online sources. Although queried DNSWLs are well known, trusted entities, it is suggested that TXT records be reported only if, upon inspection, their content is deemed actually actionable. If they contain non-ASCII characters, they need to be encoded as appropriate.

6. References

6.1. Normative References


6.2. Informative References


Appendix A. Example

Delivered-To: recipient@example.org
Return-Path: <sender@example.com>
Authentication-Results: mta.example.org;
    dkim=pass (whitelisted) header.i=@example.com
Authentication-Results: mta.example.org;
    dnswl=pass dns.zone=list.dnswl.example
    policy.ip=127.0.10.1
    policy.txt="fwd.example http://fwd.example/s?s=100"
Received-SPF: fail (Address does not pass Sender Policy Framework)
    client-ip=192.0.2.1;
    envelope-from="sender@example.com";
    helo=mailout.fwd.example;
    receiver=mta.example.org;
Received: from mailout.fwd.example (mailout.fwd.example [192.0.2.1])
    (TLS: TLSv1/SSLv3,128bits,ECDHE-RSA-AES128-GCM-SHA256)
    by mta.example.org with ESMTPS; Mon, 04 Apr 2016 23:11:24 +0200
    id 00000000005DC044.000000005702D87C.000007FC

Trace fields added at the top of the header by multiple agents at various stages during processing at the final MTA

The message went through a third party, fwd.example, which forwarded it to the final MTA. Such mail path was not arranged beforehand with the involved MTAs, it emerged spontaneously. This message would not have made it to the target without whitelisting, because:
the author domain published a strict SPF policy (-all),

- the forwarder did not alter the bounce address, and

- the target usually honors reject-on-fail, according to Section 8.4 of [RFC7208].

However, the target also implemented the last paragraph of Appendix D.3 of [RFC7208]. Rather than rejecting the message outright before DATA, the MTA received it, recorded the SPF fail result, and indicated the local policy mechanism which was applied in order to override that result. Subsequent filtering detected no malware and verified DKIM [RFC6376]. It would still have been possible to reject the message, based on its content. It is at these later stages, after receiving the body and also during delivery, that a deeper knowledge of the policy values obtained from dnswl.example can allow weighting that score against other factors.

Author’s Address

Alessandro Vesely
v. L. Anelli 13
Milano, MI 20122
IT

Email: vesely@tana.it