Appsawg
Internet-Draft
Intended status: Experimental
Expires: April 15, 2016

D. Otis
Trend Micro

October 13, 2015

Third-Party Authorization Label
draft-otis-tpa-label-08

Abstract

This experimental specification proposes a Third-Party Authorization Label (TPA-Label) as a DNS-based method that allows Trusted Domains an efficient means to authorize acceptable Third-Party Domains. This method permits autonomous unilateral authorizations and uses scalable individual DNS transactions.

A TPA-Label Resource Record transaction asserts an alignment exception to convey informally Federated Domains. It affords recipients a practical and safe means to extend Domain Alignment. Exceptions are managed by either the Trusted Domain, or their agent, seeking to avoid disruption of informal services enjoyed by their users. Third-Party Authorization of a Federated Domain eliminates a need to share private credentials.

Requirements Language

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 [RFC2119].

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 April 15, 2016.

Copyright Notice

Table of Contents

1. Introduction

A TPA-Label Resource Record supports an authorization of separately validated domains. This added authorization step avoids a need to share private credentials. Also, it ensures each domain remains apparent and open to validation when establishing an informal federation of domains protecting Federated Domain Identities. To improve security, authorization records may also limit how they are to be applied.

With TPA-Label Resource Records, mailing lists, among similar Third-Party Domain services, can indirectly assert protection of identities when the source domain is within an informally Federated Domain. Since mailing-lists receive differently formatted messages, a common practice is to convert multi-party conversations into consistent and compact formats facilitating the organization of the many multi-party conversations. Such processing often breaks any meaningful message signature. With the proposed scheme, trusting the federated message source can supersede otherwise broken alignment validation.

Trusted Domains can seek to ensure the domains they federate protect the Federated Identity. In situations where the Trusted Domain cannot be confirmed, TPA-Labels are able to signal which domains are within the Trusted Domain's informally established federation. When a user wishes to utilize an informal Third-Party Domain service, it is both logical and desirable to retain the original Federated Identity to better convey who substantially created message content. Not retaining this identity would otherwise prohibit subsequent review of prior exchanges. However, when recipients wish to determine whether to trust the Federated Identity, Domain Alignment with the validated source may not exist. TPA-Labels can indicate whether the source domain has been informally federated by the Trusted Domain.

Federating a domain also protects this domain's messages from being returned in DMARC feedback and possibly inadvertently exposing members in conversation with the Trusted Domain. Such exposure might be used to facilitate convincing phishing attacks, for example.

2. Terminology

Please see [RFC5598] for general email terminology.

The following additional terms are used:

Transparent Domain Authorization: Third-Party Domains validating as the Trusted Domain represent a type of transparent authorization.

This method normally depends on securely sharing private details between domain owners and providers. However, private sharing between different administrative domains is expensive and carries some risk a security breach may result in the wrong administration being held accountable or more resources being placed in peril.

Trusted Domain: Often a visible domain that acts as a basis for acceptance and/or subsequent actions. For DMARC, this is the fully qualified domain name found in the From header field.

Author Domain: See Section 3 of [DMARC] specifying the domain of the From header field which represents the Trusted Domain.

Federated Domain: A domain (among possibly others) working in concert with the Trusted Domain authorized as protecting Federated Identifiers.

Federated Identity: Identity protected by a Federated Domain. In the case of DMARC, this identity is contained in the From header field.

Domain Alignment: Strict alignment requires matching the Trusted Domain. Relaxed alignment allows source domains to be a sub-domain of the Trusted Domain.

Third-Party Authorization: A different domain authorized by the Trusted Domain.

Informal Third-Party Service: A Service not by the Trusted Domain that does not require administrative cooperation for users to independently establish their own access credentials.

Third-Party Service: A Service not by the Trusted Domain.

Third-Party Domain: A domain that is not the Trusted Domain.

TPA-Label Authorization: The referencing domain which meets the validation and header field content requirements of the resolved

TPA-Label Resource Record is thereby authorized by the Trusted Domain.

TPA-Label Listed Domain, TPA-LLD: TPA-Label Listed Domain, TPA-LLD, is a TXT Resource Record referenced with the hash value of the domain being authorized. This Resource Record is published within a Trusted Domain. When a "tpa" tag exists, the referencing domain (the domain used to generate the label) must be within the listed domains. When the "tpa" tag does not exist, the referenced domain is presumed. The "param" tag may stipulate a required existence of additional header fields, or indicate alternate domain validation methods to be applied against specific elements. The "param" tag may also indicate that the domain matching in the prior "tpa" list validated according to "param" associated listed methods, is specifically not federated by the Trusted Domain.

3. Domain Validation Issues

Changing the validated domain, the one referenced by SPF [RFC7208], or the one adding a DKIM [RFC6376] signature, is not a problem since it is rare for acceptance to be based on From header field Domain Alignment. However, when acceptance is based on the From header field alignment, as in the case of [DMARC] using either SPF or DKIM, this can disrupt many Third-Party Services. The disruption becomes egregious when messages from the domain's own users are rejected based on the level of this domain's asserted alignment practices. At the strictest alignment level, an erroneous assertion not only disrupts messages from their users, it can also affect subscriptions or services for other users of the Third-Party Service.

DKIM, unlike SPF, permits better retention of From/signature alignment where only the From header field could be signed. Nevertheless, the integrity of the original DKIM signature is likely affected by message flattening, inclusion of Subject tags, or appended list footer information. Just signing the From header field is not a practical solution, because it would expose this message fragment to replay abuse, even when given short signature expiry.

TPA-Label authorization may individually authorize domain validation methods. This may either increase or decrease the number of validation methods normally used. For example, a virtual server may share an IP address with thousands of different domains. Its authorization may need to exclude IP addresses as a basis for validation. In the TPA-Label scheme, unless a validation method is

asserted, no changes to the domain validation process should be assumed.

TPA-Labels can minimize the number of systems involved and the related deployment time before disruption of legitimate messaging is avoided. TPA-Labels should also ensure greater cooperation to sustain desired protections. These types of restrictions are increasingly likely to be relied upon to help mitigate harm caused by future breaches.

4. Complaince with Non-Transactional Messages

Administrative domains, that assert all of their outbound message sources can be validated as having aligned domains, offer significant forensic value. However, messages where domains are not in alignment remain a potential issue. Only domains offering messages of a transactional nature are unlikely to benefit from the use of TPA-Labels.

This document describes how any Trusted Domain publishing DMARC records can autonomously authorize other validated domains. TPA-Labels offer secondary compliance options whenever authorized exceptions are needed to permit the use of Third-Party Domains. The intended purpose of TPA-Label Resource Records is to improve acceptance rates of genuine messages, to minimize DNS use, to minimize success rates for phishing, to improve sorting protections, and to minimize a recipient's administrative costs.

TPA-Label Resource Records authorize Third-Party Domains and services to extend compliance options for asserted practices defined by [DMARC]. Domains, that both reference and are listed, and also comply with a TPA-Label resource record, should be considered equivalent to the authorizing Trusted Domain when assessing compliance with DMARC asserted practices. Otherwise, DMARC may offer non-compliant messages as feedback to foreign domains which may unintentionally expose private exchanges on behalf of the Author domain's users or those of the Federated domain.

5. TPA-Label Listed Domain, TPA-LLD,

TPA-Label Listed Domain, TPA-LLD, is a TXT resource record referenced with a TPA-Label published within a Trusted Domain. When a "tpa" tag exists within the TXT resource record located at the TPA-Label, the referencing domain (the domain used to generate the label) must be within the listed domain. When the "tpa" tag does not exist, the referenced domain is presumed listed. The "param" tag may stipulate

existence of additional header fields, or indicate alternate validation methods applied against specific email elements.

Third-Party Domain validation might use a DKIM signature or confirm the Authorized Domain using specific methods with various path related email elements. The default assertion for param is 'd' and 'm', indicating DKIM or the Mail From parameter processed by SPF confirms the Authorized Domain when no other method is specified. The 'S' and 'L' param do not confirm the domain, but requires at least one Sender or List-ID header field to hold a TPA-LLD respectively. The 'O' param can be stipulated when the Authorized Domain does not offer DMARC acceptance or validate access in association with From header fields. The 'e', 'h', and 't' indicate specific alternative methods using message elements to confirm the Authorized Domain.

When any param method is asserted (denoted by a lower case letter), methods not listed should not be considered to provide valid results. The 'n' assertion indicates that the prior domain listed in the same TPA-LLD is specifically not federated as determined by either the default or listed methods. Being compliant with TPA-LLD allows the referencing domain to informally act on behalf of the Trusted Domain. Indicating domains as not federated necessitates the use of sequence sensitive "tpa" and "param" pairs within the TPA-LLD. Per [RFC5321], domain name comparisons, as well as TPA-Labels, are case insensitive.

6. TPA-Label Resource Record Authorization Considerations

When a Trusted Domain is not within a DKIM or SPF validated domain, the TPA-LLD scheme can extend Domain Alignment compliance. The TPA-LLD scheme with an 'S', or 'L' param requires the respective Sender header field or a List-ID identifier of the List-ID header field to exist for at least one of the params, and to contain a domain within the TPA-LLD for authorization to be valid. The 'd' param permits validations based upon the DKIM signing domain. The 'm' param permits validations based upon the return path (Mail From) domain. The 'e', 'h', and 't' params permit acceptance based upon validation of the client hostname (EHLO/HELO).

The 'S' and 'L' params support message sorting. Any matching header field with a domain within the TPA-LLD allows recipients to differentiate sources, which satisfies requirements for any other 'S' or 'L' param. The 'S' and 'L' params provide Trusted Domains a means to limit domain authorizations.

The TPA-LLD scheme plays the role of only qualifying acceptable domains with the goal of improving delivery acceptance, such as

messages from specific mailing-lists. The TPA-LLD authorization scheme only requires that DNS publications be made by the Trusted Domain, even when the sending domains and the Trusted Domain differ. This approach eliminates a need to exchange private information thus protecting the domain's integrity. Before TPA-LLD authorization is deployed, the Trusted Domain should be assured by domains being authorized that appropriate measures are in place to validate those submitting messages and ensure the Federated Identity is protected.

Retaining validation and authorization for the From, Sender, and List-ID header fields, and being able to ensure Third-party inclusion of a Sender or List-ID header fields, enhances protections afforded by message sorting. This protection reduces susceptibility to deceptive look-alike phishing attempts. Use of subdomains that assert less stringent practices might inadvertently combine with those having more stringent practices when sorting is based upon parent domains. Consistently using the same domain prevents possible confusion that could be exploited to deceive recipients.

TPA-Label authorization will not ensure all possible spoofing is prevented. However, by permitting broader use of strict alignment practices, this should generally reduce the level of spoofing over what might be otherwise allowed. Authorized third party messages SHOULD NOT receive annotations that indicate the message contains validated identities. The TPA-LLD param SHOULD include the 'S' or 'L' param where appropriate to allow recipients a means to isolate and distinguish different message sources.

7. Evaluating the Third-Party Domain

A Trusted Domain deploying a TPA-Label Resource Record does so on a trust basis. Reasons for deploying TPA-Label Resource Records might be to allow deployment of more stringent DMARC records while also utilizing Third-Party Services.

When an authorized Third Party domain does not employ DKIM or SPF or does not include Authentication-Results header fields [RFC7001] or perhaps [I-D.kucherawy-original-authres] (OAR) or its "X-" version could allow authorizations to be exploited. For Third Party domains not applying DMARC but capture the OAR, past compliance with DMARC based on the OAR can be made a requirement for authorization.

While conceivably Domain Alignment might just rely on the content of the Original-Authentication-Results header, whether to trust this, or any other message content can not be based on the mere acceptance of the message alone. Whether false content even effects message

acceptance would be difficult to determine. Only the Trusted Domain is able to make this type of determination based on their knowledge of outbound messages and corrections needed based on DMARC feedback which they then share in the form of a TPA-Label.

7.1. Third Party Authorization - Closed Mailing List Example

The Trusted Domain wants to deploy a TPA-Label Resource Record for a mailing list with a closed posting policy. The mailing list redistributes email which breaks the Trusted Domain Alignment, and the mailing list offers a means to validate the mailing list domain and includes an Authentication-Results header field for posted messages. The closed posting policy can be enforced by requiring subscribers to validate control of their Author Addresses by responding to encoded "pingback" email sent to these addresses.

Since the mailing list validates their domain as indicated in the TPA-Label, and validates control of the posted message Author Address, and includes Authentication-Results header fields, and includes a List-ID header field, the referenced TPA-Label Resource Record can include an 'L' param value to stipulate that the Third-Party Domain messages contain an authorized List-ID domain.

7.2. Third Party Authorization - Open Mailing List Example

The Trusted Domain wants to deploy a TPA-Label Resource Record for a mailing list with an open posting policy. The mailing list redistributes email in a way that breaks Trusted Domain alignment, does not post from an Author Address not in compliance with DMARC, offers a means to validate the mailing list domain, and it includes an Authentication-Results header field for posted messages.

Since the mailing list validates the domain as indicated in the TPA-Label, and is configured to include Authentication-Results header fields and possibly the Original-Authentication-Results [I-D.kucherawy-original-authres], and includes a List-ID header field, the referenced TPA-Label Resource Record can include an 'L' param value to stipulate the Third-Party Domain messages contain an authorized List-ID domain.

7.3. Third Party Authorization Example - Sender Header Field

Trusted Domain "example.com" wishes to temporarily employ the service agency "temp.example.org" to handle overflow secretarial support. The agency "temp.example.org" sends email on behalf of the executive staff of "example.com" and adds the Sender header field of "secretary@temp.example.org" in the email.

Since "temp.example.org" only allows its own staff to email through its server which adds "temp.example.org" DKIM signatures, a TPA-LLD can include the "temp.example.org" domain with an 'S' and 'd' param to specifically authorize DKIM signed messages containing the Sender header field, to help ensure these messages are not handled as phishing attempts.

7.4. Services Lacking DKIM Signatures

7.4.1. Abuse and DSN Reporting

There is likely little interest for an otherwise uninvolved domain to receive a massive number of bogus messages being returned as feedback. Often the purpose of feedback is to discover compromised systems or accounts actively being exploited in some manner. Unless the Trusted Domain is confirmed as having handled or authorized the handling of the message, only statistics and samples should be reported to the associated Autonomous System [RFC1930], and perhaps to the Trusted Domain when interest is expressed.

The 'd', 'e', 'h', 'm', and 't' param options within the TPA-LLD records allow the Trusted Domain to be associated through various methods. In this case, appropriate DSN or abuse reporting to the Trusted Domain is better assured as well.

7.4.2. Third Party Authorization Example - SMTP Host

Trusted Domain "example.com" makes use of invite services. This service does not utilize DKIM, where the host name given by the EHLO command is "invite.example.net". The Trusted Domain can authorize the domain "invite.example.net" or "example.net" with the param of 'e' to improve acceptance of messages that are sent on behalf of "example.com" from this outbound server.

7.4.3. Third Party Authorization Example - Return Path

Trusted Domain "example.com" makes use of tell-a-friend services. This service does not utilize DKIM with its own return path as "customer@taf.example.net" in the SMTP exchange. The Trusted Domain can authorize the domain "taf.example.net" with the param of 'm' to improve acceptance of messages that are sent on behalf of "example.com" from this outbound server.

7.4.4. Use of Path Authorization

Those using validations related to 'e', 'h', 'm' param options should

not authorize domains requiring more than an average number of network transactions. Those implementing DMARC should also limit the number of DNS transactions attempted, otherwise this could negatively impact unrelated domains when evaluating path related validation.

> Methods that create subsequent transactions based upon the macro expansion of email-address local-parts should not be used. Libraries that process SPF [RFC7208] record scripts may invoke a large number of DNS transactions from cached records, and target unrelated domains with queries modulated by the local-part component through receiver macro expansion.

8. DNS Representation

The receiver obtains domain authorizations with a DNS query for an IN class TXT TPA-Label Resource Record located below the "_smtp._tpa.<Trusted-Domain>" location. The TPA-Label itself is generated by processing the domain in question, which normally matches the DKIM signature's "d=" parameter. The Trusted Domain provides authorization for other domains with the existence of a TPA-Label TXT resource record. When a "tpa" tag value exists, it MUST include the referenced domain before authorization is valid. This represents an informal authorization on behalf of the Trusted Domain which can be limited by the "param" tag value for specific message elements.

A Trusted Domain may wish to delegate the listing of Third-Party Services to a different administrative domain. Ideally, this would be accomplished by delegating the _tpa.<Trusted-Domain> zone to the administrative entity handling publication of TPA-Label Resource Records. This delegation could also be done unilaterally with a DNAME [RFC6672] resource record published at _smtp._tpa.<Trusted-Domain>.

Character-strings contained within the TXT resource record are concatenated into forming a single string. A character-string, as defined in [RFC1035] Section 3.3 for resource records, is a single length octet followed by that number of characters treated as binary information.

The TPA-Label Resource Records should be located at these domains:

> <TPA-Label>._smtp._tpa.<Trusted-Domain>.

9. TPA-Label and Tag Syntax Definitions

Augmented BNF for Syntax Specifications:

```
asterisk = %x2A ; "*"
dash = %x2D ; "-"
dot = %x2E ; "."
underscore = %x5F ; "_"
ANY = asterisk dot ; "*."
dns-char = ALPHA / DIGIT / dash
id-prefix = ALPHA / DIGIT
label = id-prefix [*61dns-char id-prefix]
sldn = label dot label
base-char = (dns-char / underscore)
domain = *(label dot) sldn

FWS       =  ([*WSP CRLF] 1*WSP) ; omits RFC5322 obs-FWS
tag-sep   =  %x3B ; "%"
tag-list  =  tag-spec 0*( tag-sep tag-spec ) [ tag-sep ]
tag-spec  =  [FWS] tag-name [FWS] "=" [FWS] tag-value [FWS]
tag-name  =  ALPHA 0*ALNUMPUNC / "v" / [["tpa"] / ["param"]]
tag-value =  [ tval 0*( 1*(WSP / FWS) tval ) ]
                ; WSP and FWS prohibited at beginning and end
tval      =  1*VALCHAR
VALCHAR   =  %x21-3A / %x3C-7E
                ; EXCLAMATION to TILDE except SEMICOLON
ALNUMPUNC =  ALPHA / DIGIT / "_"
```

10. TPA-Label Generation

The TPA-Label is generated by nesting functions as follows:

"base32" function is defined in [RFC4648].

"sha1" function is defined in [FIPS.180-2.2002].

"lcase" converts upper-case ALPHA characters to lower-case.

"tpa-domain" is normally the "d=" tag value defined in Section 3.5 of [RFC6376].

(underscore) base32(sha1(lcase(tpa-domain)))

The TPA-Label is created from the hash value returned by the "sha1" function of the tpa-domain expressed in lower case ASCII. Any terminating period is not included with the tpa-domain, as indicated by the ABNF definition.

Note: No newline character, 0x0A, is to be appended to the end of

the domain name, as might occur with the command line generation of sha1 values. For example, these command line appended newlines can be avoided by using the 'echo -n" option.

The label encoding process inputs the hash as a byte stream of four 40-bit data blocks where each data block outputs 8 encoded characters. Proceeding from left to right, a 40-bit input group is formed by concatenating 5 bytes. The 40-bit input is then treated as 8 concatenated 5-bit groups, each of which is translated into a single digit of the base32 alphabet. The bit stream is ordered with the most-significant-bit first, being the high-order bit of the first byte. The entire output is then concatenated first to last, left to right, into 32 characters prefixed with an underscore.

11. TPA-Label TXT Resource Record Structure

Every TPA-Label TXT resource record MUST start with the version tag, so the first six characters of the record are lowercase "v=tpa1", TPA-Label syntax descriptions for additional tags follow the tag-value syntax described in the ABNF below, the WSP token is inherited from [RFC5322]. The ALPHA and DIGIT tokens are imported from [RFC5234]. The "param" values refer only to domains previously listed in the TPA-LLD which makes the "tpa" and "param" tags sensitive to their sequence.

The tags used in TPA-Label Resource Records are as follows:

Tag	Function
v	Label Version (version-tag)
tpa	Authorized Domains List (tpa-tag)
param	Authorization Param List (param-tag)

TPA-Label Tags

param values	Field or Parameter	Method
L	List-ID Header Field	Match List-ID Identifier
S	Sender Header Field	Match Address Domain
O	Original Authentication Results Header Field	Match Address Domain
d	DKIM Signature	Match Signature Domain
e	SMTP Hostname	Resolve Hostname IP Addr
h	SMTP Hostname	Pass SPF with Hostname
n	Not Federated	See Other Params
m	MailFrom	Pass SPF with MailFrom
t	SMTP Hostname	Cert of Hostname

TPA-Label Param Values

12. TPA-Label Resource Record Definition

Tags in the TPA-Label Resourse Record are shown below. The ver-tag MUST be present as the left most tag. Unrecognized tags MUST be ignored.

TPA-Label Resource Record Definition

```
tpalabelrr = v-tag [tag-sep] 0*( 1*(WSP) tag-list) ]
```

13. TPA-Label Resource Record Version

Label Version (Required). This tag defines the version of the TPA-Label. Only recognized tpa:param values offer DMARC authorizations (except for param "n" which excludes domains validated per other param values).

"v" tag

```
v-tag = %76.3d.74.70.61.31 ; "v=tpa1"
```

14. Authorized Validated Domains

Authorized validated domain list. (optional) This tag, when present, MUST contain a domain that repeats all or right-most portions of the domain encoded within the TPA-Label Resource Record. This option ensures the proper handling of possible hash collisions. When a domain is prefixed with the "*." ANY label, then all subdomains of this domain are to be considered included within the list. When the 'tpa' tag is not present or has no value, it should be assumed to compare with the domain used to generate the TPA-Label. The purpose of the ANY label is to reduce the size of the resource records. Containing the entire string to confirm hostnames or List-ID content is unnecessary. The hash label must still be an exact match of the domain authorized. Additional domains may be included as optional Sender or List-ID comparison options. The tpa list is optionally followed by param list. There can be multiple tpa:param sets.

Use of the ANY label is not intended to support wildcards for referencing hash labels. No wildcard labels are to be used below the "_tpa." label to access DNS resources.

"tpa" tag

```
ad_val = [ANY] domain
ad-list = %x74.70.61 *WSP "=" [ ad_val 0*( 1*(WSP) ad_val )]
```

14.1. TPA-Label Resource Record Param Syntax

Authorization Param List (Optional). This tag defines a list of assertions for the preceding (if listed) domains which indicate various email-address locations within the message and authorized validation methods. Only recognized param values offer any form of DMARC authorization. The "n" param however excludes the prior tpa list as not being within the federation.

"param" tag

```
pa_val = "L" / "S" / "O" / "d" / "e" / "h" / "m" / "n" / "t"
pa-list = %x73.63.6f.70.65 *WSP "=" [ pa_val 0*( 1*(WSP) pa_val )]
```

14.2. Header Dependent Authorizations

14.2.1. List-ID Header Field

The "L" param asserts that authorization is valid only when a List-ID identifier of the List-ID header field [RFC2919] contains a domain that is within a domain listed in the TPA-LLD "tpa" tag.

The syntax of the List-Id header field is as follows:

```
list-id-header = "List-ID:" [phrase] "<"identifier">"CRLF
```

14.2.2. Sender Header Field

The "S" param asserts that authorization is valid only when the domain in the Sender header field is within the TPA-LLD.

14.2.3. OAR Header Field

The "O" param asserts that authorization is valid only when the domain in an Original Authentication Results header field indicates it passed based on a domain within the TPA-LLD or the Trusted Domain itself.

14.2.4. Combined 'L' or 'S' Params

When combined, the params 'L' and 'S' require that either a List-ID identifier of the List-ID header field or the Sender header field must contain a domain within the TPA-LLD for the authorization to be valid.

14.3. DKIM signed domain

14.3.1. DKIM signed

The "d" param asserts that messages carrying the Trusted Domain within the From header field are authorized to be signed by the TPA-LLD.

14.4. SMTP Host domains

The "e" param asserts that host names given in [RFC5321] EHLO or HELO commands within TPA-LLD is authorized when the hostname resolves the server's IP address.

14.5. SMTP Host domains

The "h" param asserts that host names given in [RFC5321] EHLO or HELO commands within TPA-LLD is authorized only when this hostname submitted to an SPF [RFC7208] process returns pass.

14.6. MailFrom Parameter

The "m" param asserts that an email-address domain in the [RFC5321] MAIL command within a TPA-LLD is authorized only when this email-

address submitted to an SPF [RFC7208] process returns pass.

14.7. Not Federated

The "n" param asserts that a previous "tpa" listed domain is specifically not federated and not authorized. The use of this parameter is to suppress subsequent processing that may otherwise be needlessly repeated in the case where the third-party is being detected as conveying user messages but are not authorized for policy related reasons, such as not adequately protecting the Federated Identity.

14.8. SMTP Host domains

The "t" param asserts that host names given in [RFC5321] EHLO command after [RFC3207] negotiation where the Cert DNS-ID domain is within TPA-LLD is authorized. It will also be interesting to see whether [I-D.ietf-dane-smtp-with-dane] establishes a way to authenticate sending domains.

Note to RFC Editor: Remove this comment before publishing.
Currently, no general practice employs certificates to confirm the domain of the client initiating a connection. This may be needed for clients within IPv6 IP address space where tunneling, carrier grade NATs, and rapid space assignment without any practical reverse mapping reduces the effectiveness of IP address based reputations.
There is an existing TLS option for SMTP and an ongoing effort to standardize automated server confirmation. It might be possible to leverage this effort to establish practices used at the client. See conversations defined in [RFC4954] Section 4. For information related to ongoing server related efforts see:
[RFC6125] and [RFC6698]

15. TPA-Label Resource Record Query Transactions

The discovery of TPA-Label Resource Records need not be subsequent to the discovery of the DMARC record. However, when no DMARC record is discovered which includes the tag value of "tpa", the verifier MAY assume no TPA-Label Resource Records have been published. Otherwise, when there is no Trusted Domain validation, the discovery of TPA-Label Resource Records should be attempted.

16. TPA-Label Resource Record Compliance Extension

The signing practice compliance assessment of Third Party Signatures is a discretionary operation performed by the verifier. For messages that do not have valid Trusted Domain alignment, a verifier may decide to assess compliance for Third Party messages when there is a DMARC tag of "tpa". Elements then referenced in the TPA-Label param values of "d", "m", "e", "h", "t" are to be checked in their listed succession. One of the following sets of conditions MUST be met for the result to be considered a pass:

For Third Party DKIM signatures, the following represents the set of conditions to be checked:

- The Third Party Signature MUST validate according to [RFC6376].
- A TXT resource record, referenced by a TPA-Label created by the DKIM signature "d=" tag, MUST exist in DNS.
- The discovered TPA-Label Resource Record structure MUST be valid.
- The domain that created the TPA-Label MUST be within the TPA-LLD.
- Where a param of 'd' is specified, the Trusted Domain MUST have an authorized DKIM signature.
- Where a param of 'L' or 'S' is specified, a List-ID identifier in the List-ID header field or a Sender header field MUST contain a domain within the TPA-LLD. This provides Third-Party services a reason to ensure their outbound messages do not spoof these associated header fields.
- Where a param of 'O' is specified, an Original Authentication Results header field MUST indicate a pass for the Trusted Domain or for a domain within the TPA-LLD. This parameter requires the message was received from an approved Originating source.

For non-DKIM validations, the TXT record discovery process continues until a TPA-Label Resource Record structure is found where:

One of the three possible TXT resource records is checked in their listed succession. Each would be referenced by an 'h' or 'e' or 't' related domain given by [RFC5321] EHLO or HELO command, this domain with left-most label omitted, or by an 'm' related email-address domain within the [RFC5321] MAIL command.

- The discovered TPA-Label Resource Record Structure is valid.
- The domain that created the TPA-Label is within the TPA-LLD.
- The domain that created the TPA-Label corresponds with a listed param of 'e', 'h' or 'm' or 't'.
- Where a param of 'L' or 'S' is specified, either the domain in List-ID given by [RFC2919] in the List-ID header field is within the TPA-LLD, or a Sender header field contains a domain within the TPA-LLD respectively.
- Once these four conditions have been met, for 'h' or 'm' params

the domain MUST be confirmed by submitting the domain to an SPF process that then returns pass. The 'e' param MUST be confirmed by a forward DNS reference that resolves the address of the SMTP client. The 't' param MUST be confirmed by the DNS-ID in the client certificate.

When the TPA-Label Resource Record can not be retrieved due to some error that is likely transient in nature, such as "SERVFAIL" for example, the result of the TPA-Label Resource Record compliance assessment is "temperror".

When the TPA-Label Resource Record retrieval returns a DNS "NOERROR", but not with a single record, the result of the TPA-Label Resource Record compliance assessment is "permerror".

When the TPA-Label Resource Record can not be retrieved with a DNS "NXDOMAIN" response, the result of the TPA-Label Resource Record compliance assessment is "nxdomain".

17. Alternative Mitigation Strategies

17.1. Proposed DKIM-Delegate Signature survives all Message modifications

When a domain sends a message to services likely to invalidate DKIM signatures, such as that of a mailing-list, some envision use of a [I-D.kucherawy-dkim-delegate] header field as a type of DMARC disruption prophylactic conveying signed Third-Party Authorizations similar to that of TPA-LLD.

The DKIM-Delegate header field is not a DKIM message signature nor can it generally authorize trustworthy sources. This header field represents a signed domain authorization list placed in the header field's 't' tag. To limit its duration, short expiry can be specified in the 'x' tag. Nevertheless, validity of this header field is completely independent of the message body or any other message header field.

Since DKIM-Delegate authorization is actually unrelated to any specific message, it offers less protection at the expense of a higher transactional overhead compared to that of the smaller and simpler TPA-LLD transactions. The TPA-LLD resource also better facilitates timely authorization retractions. Before DKIM-Delegate can be as effective, separate and much larger signature resources would be needed for each authorized domain to match the timely de-authorization selectivity the TPA-Label scheme permits.

Since either scheme is primarily aimed at circumventing DMARC compliance issues, once adopted, changes to DMARC related validation can quickly bolster DKIM/DMARC legacy shortfalls in a similar fashion to that offered by TPA-LLD. It is even conceivable, that to obtain an automated authorization expiry, a TPA-LLD could signal when a DKIM-Delegate should be included. The TPA-LLD can still reduce the authorization latitude offered by DKIM-Delegate. Unlike TPA-LLD, DKIM-Delegate is unable to stipulate an additional set of required conditions.

Since mailing-lists are normally fairly public, bad actors only need to subscribe to obtain a fresh set of DKIM-Delegate header fields as a means to circumvent its expiry feature. Using freshly minted DKIM-Delegate header fields also avoids elements that might be identified as pertaining to a specific campaign at little cost to the malefactor.

17.2. Vouch-By-Reference

VBR uses a third-party domain referenced in a message header to vouch for the types of messaging expected from a domain verified as part of normal message handling. Since any domain other than the DMARC domain attempting to guard against From header field spoofing is unable to make such assessments, for the purposes of Anti-Phishing, VBR lacks the necessary input and knowledge to offer timely and accurate advice. The nature of Phishing abuse is often too low in frequency for typical Anti-Spam policies to be effective.

18. Privacy Considerations

Unless all valid Third-Party Domains have been authorized, personally identifiable information will be exchanged within the DMARC feedback. This feedback can unintentionally expose private exchanges made on behalf or the DMARC domain's users. To the greatest extent possible, this feedback information should not be shared with other domains not offering the information. This feedback can even identify mailing-list subscribers that never sent any message to the list, or invoices made on behalf of an accountant's client.

As with other authorization schemes that utilize DNS, relationships are publicly revealed. This is the nature of SPF authorization which reveals first party services being used. A TPA-Label on the other hand can resolve a hash obscured Third-Party Service. Unlike SPF, a TPA-Label does not include any user identity related parameters and does not reveal any users specific relationships. In addition, these relationships are accessed with a hash of the entire domain. Use of a few random subdomains can inhibit discovery of these relationships.

However, the low latency of DNS means resource records can not be assumed to remain secret.

Even so, disclosures of Third-Party Services might be justified by dissuading malefactors who have compromised the Trusted Domain and then are able to subsequently spoof the discovered personal relationships. Such spoofing might be seen as causing greater harm than public knowledge of possible Third-Party Services used by the Trusted Domain's users.

It seems this can not be overstated: The overhead associated with managing a "_tpa." zone is fairly small and is well offset by squelching DMARC feedback generation and the remediation of a loss of legitimate messages. Alternatives to TPA-Labels are likely to be the dissemination of plaintext lists of domains known to cause alignment failures, although operating in full compliance with SMTP protocols and practices. The dissemination of lists lessens the domain's privacy, and their ability to react to mitigate abuse.

19. IANA Considerations

19.1. Moving RFC6541 to historic

This document is seeking to replace [RFC6541] and to move it to historic.

19.2. TPA-Label (TPA-LLD) Parameters

To accommodate the extensions to [RFC7001] needs the following elements to be added:

Type	Reference
tpa	(this document)

TPA-Label Resource Record validation Method

19.3. Email Authentication Method Registry

To accommodate the method derived from TPA-Label Resource Record processing, the IANA Registry "Email Authentication Method" defined by Section 6.2 of [RFC7001] needs the following elements to be added:

Method	Defined	ptype	property	value
tpa-lld	(this document)	domain	3p-dom	Domain evaluated. The method results from [RFC7001] should also be included in a Authenticated Results header field.
			param	value of param (Section 19.6) tag. (When 'param' contains 'e', 'h' or 'm', the iprev [RFC7001] (Section 3) method results should also be included in the Authenticated-Results header field to capture the SMTP client IP address.
			ca-param	The params (Section 19.6) with a compliance assessment as pass
			tpa	Value of tpa (Section 14) tag at time of compliance assessment

TPA-Label Resource Record validation Method

19.4. Email Authentication Result Names Registry

To accommodate the results derived from TPA-Label Resource Record processing, the IANA Registry "Email Authentication Method" defined by Section 6.3 of [RFC7001] needs the following elements added:

code	method	meaning
none	tpa-lld	No TPA-Label was published
pass	tpa-lld	Section 16
tempfail	tpa-lld	Section 16
permfail	tpa-lld	Section 16
hdrfail	tpa-lld	The TPA-Label Resource Record param values of "S" or "L" failed to match.
nxdomain	tpa-lld	When obtaining the TPA-Label Resource Record, DNS indicated this domain does not exist.

TPA-Label Resource Record complaince assessment Results

19.5. Third Party Authorizations Labels Registry

Names of tags that are valid in TPA-Label Resource Records with the exception of experimental tags Section 11 MUST be registered in this created IANA registry.

New entries are assigned only for values that have been documented in a published RFC that has had IETF Review, per IANA CONSIDERATIONS [RFC5226].

Each tag registered must correspond to a definition.

The initial set of values for this registry is:

tag	defined	definition
v	Section 11	Label Version
tpa	Section 14	List of Authorized Domains or Identifiers
prior tpa param	Section 14.1	Section 19.6

TPA-Label Resource Record compliance assessment Results

19.6. Third Party Authorizations Param Registry

Values that correspond to Section 14.1 MUST be registered in this created registry:

New entries are assigned only for values that have been documented in a published RFC that has had IETF Review, per IANA CONSIDERATIONS

[RFC5226].

Each value registered must correspond to a definition.

The initial set of values for this registry is:

value	defined
L	Section 14.2.1
S	Section 14.2.2
O	Section 14.2.3
d	Section 14.3
h	Section 14.5
e	Section 14.4
m	Section 14.6
n	Section 14.7
t	Section 14.8

TPA-Label Resource Record compliance assessment Results

20. Security Considerations

This draft extends Domain Alignment validation practices that depend on DKIM [RFC6376] or SPF [RFC7208]. Most related security matters are discussed in those specifications. Additional considerations are also included in [RFC6377]. Security considerations for the TPA-LLD scheme are mostly related to attempts on the part of malefactors to falsely represent themselves as others, often in an attempt to defraud either the recipient or the alleged originator. Some receivers mistakenly bypass validation of the [RFC5322] header fields because a signature from a Trusted Domain had been confirmed as perhaps suggested in [RFC5863]. Do not omit the validation of header fields unless the message is not accepted for other reasons.

Additional security considerations regarding DKIM signing practices may be found in the DKIM threat analysis [RFC4686].

20.1. Benefits to Recipients

The verifier, after validating a Federated Domain, will have significantly greater confidence in the Third-Party, than when no TPA-Label Resource Record is obtained. This enhanced confidence may, at the receivers' discretion, cause a message to be delivered to the recipient with less stringent assessments.

20.2. Risks to Recipients

The decisions a recipient makes in regard to message filtering based on TPA-Label Resource Records are likely to depend on the system integrity of the Third Party with respect to the validation methods determined by authorization param labels. When the 'e', 'h', or 'm' param domain is not confirmed, or the Third-Party Domain does not validate the submitter, there is a risk of accepting potentially spoofed messages. Authentication-Results header fields then play an important role when there is no out-of-band validations confirming the submitter. Without proper Authentication-Results handling by the Third-Party, there is also risk of accepting potentially spoofed messages.

With the TPA-Label specification, third party validation provides verifiable value. Implementers should consider the possibility a malefactor will send a message having a large number of valid DKIM Signatures. Verifying all the signatures may consume a large amount of processing resources. As such, it might be worth checking for the existence of a TPA-Label Resource Record first to minimize network amplification concerns. Section 15 describes a quick check to see if TPA-Label Resource Records may exist. Additionally, validating DKIM signatures and obtaining related resource records might be limited to known trustworthy domains.

Services that depend only upon path authorizations might permit the Trusted Domain to be spoofed and yet obtain acceptance. During such events, the Trusted Domain might need to retract its authorization from the service. For this reason, path related validation based on IP addresses should only be used as a carefully monitored interim solution.

20.3. Benefits to Trusted Domains

TPA-Label Resource Records can replace domain delegations, selector/key record mirroring, or key exchanges. A significant number of details are associated with selector/key records. These details include user limitations, suitable services, key resource record's Time-To-Live, revocation and update procedures, and how the DKIM Signature header field's 'i=' semantics are to be applied. In addition, services that depend upon DKIM keys are better secured by not delegating these DKIM keys, where instead the TPA-LLD scheme allows Trusted Domains an ability to limit the scope of their authorizations, while also not being mistaken for having validated the entity submitting the message.

TPA-Label Resource Records convey which domains are authoritative even when they are not the Trusted Domain. However, Authorized

Domains are unable to utilize the DKIM signature's 'i=' semantics to directly assert which identifiers on whose behalf a signature was added. As such, no domain should be authorized unless it is trusted to ensure the Federated Identity of an email undergoes validation that offers acceptable protections for the Trusted Domain. For example, such validation might ensure submitting entities have demonstrated receipt of "pingback" messages sent to the Federated Identity (Author's address) contained within the messages being signed.

By deploying TPA-Label Resource Records, Trusted Domains benefit when recipients assess the senders' practice compliance by using the TPA-LLD scheme. These recipients will be less likely to drop the Trusted Domain's genuine messages, whenever the Trusted Domain attempts to restrict acceptance. Restricting acceptance of non-compliant messages is the basic motivation for publishing DMARC records. In addition, recipients are more likely to validate messages by an Authorized Domain.

Broader use of strict DMARC alignment assertions provides a greater likelihood of being able to eliminate a broader range of non-compliant messages, in addition to improving acceptance from authorized sources. TPA-Labels also allow Trusted Domains to control message Sender and List-ID attributes, to exclude problematic validation methods or include others as they become available.

Trusted Domains having good reputations might extend limited compliance assessment resources to otherwise unknown domains or SMTP Clients that are referenced by their TPA-LLD. Conversely, TPA-LLD resources that assert a domain as not being federated can be used to suppress any processing that might be otherwise needlessly repeated.

Privacy is better protected when messages from Federated domains won't contribute to a collection of messages as Feedback returned using less secure methods or that exposes the Federated members of a conversation.

20.4. Risks to Trusted Domains

As indicated in Section 7, it is ultimately an issue of trusting the Third Party Domain to do the right thing and not generate, or allow others to generate, messages that falsely appear to be from the Trusted Domain. The validation methods in place for different email elements need to be carefully reflected in the "param" tag of the TPA-LLD.

Authorization of mailing lists with TPA-LLD could cause a loss of confidentiality in mailing list participation by the Trusted Domain.

This might help malefactors deduce which subscription related email the Trusted Domain may receive. Because of the hashing function in generating the TPA-Label, anyone wishing to discover which domains are being authorized, has to probe each TPA-Label based on the exact domain. In addition, service organizations or community groups are able to share comprehensive lists. Such possible sharing means even though a domain has been authorized, that in itself does not mean the Trusted Domain is exchanging messages with the Authorized Domain.

20.5. Benefits to Third Party Signers

Third Party Signers benefit by allowing those using their service, the autonomy to authorize their service without needing to exchange DKIM key related details. This is particularly useful for mailing lists.

20.6. Risks caused by Third Party Signers

As mentioned before, Authorized Third Party Signers need to validate messages from Trusted Domains. This validation provides a safety mechanism for the Trusted Domain and their recipients. The Third Party may not be aware of the validation value or the message elements involved, and as a result make changes without understanding the impact this may have on Trusted Domains and their recipients. For example, the Third Party might stop DKIM signing or stop applying Authentication-Results header fields. The unexpected exposure that this might enable could allow abuse and prove detrimental for both the Trusted Domain and their recipients.

20.7. SHA-1 Collisions

The use of the SHA-1 hash algorithm does not represent a security concern. The hash simply ensures a deterministic domain-name size is achieved. Unexpected collisions can be detected and handled by using the extended TPA-Label Resource Record "tpa=" option. The use of TPA-Label Resource Records without the TPA-Label "tpa=" options does present an opportunity for an adversary to attempt to find a hash collision. Message spoofing outside the realm of DKIM protection is likely easier to achieve than finding hash collisions. There is minimal risk of TPA-Labels colliding. Listing 3 x 10^45 domains has less than a 0.1 percent risk of any two domain labels colliding.

20.8. DNS Limits

Use of the TPA-Label Resource Records, rather than simply listing the Authorized Domain, ensures the DNS record size is independent of the Third Party Domain. The typical domain name size has been steadily increasing. This increase has been caused by domain names that

encode international character sets. Perhaps, soon there will be a further increase spurred by an expanse of TLDs having larger international labels.

The maximum domain name size allowed, per [RFC1034] Section 3, is 255 bytes (or octets). Each label has a byte for its length. Every domain name adds an additional byte by having a right-most label that represents the root "." signified as a zero length label. A labeling scheme that combines together a listed domain with the publishing domain separated by some label for this convention, reduces the maximal domain name in half, where the convention label reduces this further.

If "_smtp._tpa." were used as the convention label with a simple listing method, the maximum domain name size this supports would be 128 bytes. The suffix for TPA-Labels is "_smtp._tpa." which consumes 11 bytes. The TPA-Label itself consumes 34 bytes. A domain that publishes the TPA-Labels in its domain would then have 122 bytes available for their Trusted Domain. This permits the authorization of any domain having a valid length with a deterministic amount of space available for resource records.

Normally, DNS messages should not exceed 512 bytes as per Section 2.3.4 of [RFC1035]. Using TPA-Label Resource Records in the DNS, as described by this document, consumes a consistent 50 bytes, in addition to the domain name publishing the TPA-Labels. With this being constant, a limit can be determined as a constraint to resource record size, to ensure a response does not exceed the maximum DNS message size. DNS servers that add additional resource records, for nameservers as an example, will further reduce available resource record capacity. Domains publishing TPA-Labels exceeding the DNS message limit will need to rely on recipients using TCP for DNS retrieval, or EDNS0 [RFC6891] for extended DNS lengths.

21. Acknowledgements

Daniel Black, Jeff MacDonald, Michael Deutschmann, Frank Ellermann, Murray Kucherawy, Wietse Venema, Alessandro Vesely, and John Leslie.

22. References

22.1. Normative References

[FIPS.180-2.2002]
National Institute of Standards and Technology, "Secure Hash Standard", FIPS PUB 180-2, August 2002, <http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf>.

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

[RFC2919] Chandhok, R. and G. Wenger, "List-Id: A Structured Field and Namespace for the Identification of Mailing Lists", RFC 2919, DOI 10.17487/RFC2919, March 2001, <http://www.rfc-editor.org/info/rfc2919>.

[RFC3207] Hoffman, P., "SMTP Service Extension for Secure SMTP over Transport Layer Security", RFC 3207, DOI 10.17487/RFC3207, February 2002, <http://www.rfc-editor.org/info/rfc3207>.

[RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, DOI 10.17487/RFC4648, October 2006, <http://www.rfc-editor.org/info/rfc4648>.

[RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <http://www.rfc-editor.org/info/rfc5234>.

[RFC5321] Klensin, J., "Simple Mail Transfer Protocol", RFC 5321, DOI 10.17487/RFC5321, October 2008, <http://www.rfc-editor.org/info/rfc5321>.

[RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, DOI 10.17487/RFC5322, October 2008, <http://www.rfc-editor.org/info/rfc5322>.

[RFC5598] Crocker, D., "Internet Mail Architecture", RFC 5598, DOI 10.17487/RFC5598, July 2009, <http://www.rfc-editor.org/info/rfc5598>.

[RFC6125] Saint-Andre, P. and J. Hodges, "Representation and Verification of Domain-Based Application Service Identity within Internet Public Key Infrastructure Using X.509 (PKIX) Certificates in the Context of Transport Layer Security (TLS)", RFC 6125, DOI 10.17487/RFC6125, March 2011, <http://www.rfc-editor.org/info/rfc6125>.

[RFC6376] Crocker, D., Ed., Hansen, T., Ed., and M. Kucherawy, Ed., "DomainKeys Identified Mail (DKIM) Signatures", STD 76, RFC 6376, DOI 10.17487/RFC6376, September 2011, <http://www.rfc-editor.org/info/rfc6376>.

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

[RFC7001] Kucherawy, M., "Message Header Field for Indicating Message Authentication Status", RFC 7001, DOI 10.17487/RFC7001, September 2013, <http://www.rfc-editor.org/info/rfc7001>.

22.2. Informative References

[DMARC] Kucherawy, M., Ed. and E. Zwicky, Ed., "Domain-based Message Authentication, Reporting, and Conformance (DMARC)", RFC 7489, DOI 10.17487/RFC7489, March 2015, <http://www.rfc-editor.org/info/rfc7489>.

[I-D.ietf-dane-smtp-with-dane] Dukhovni, V. and W. Hardaker, "SMTP security via opportunistic DANE TLS", draft-ietf-dane-smtp-with-dane-19 (work in progress), May 2015.

[I-D.kucherawy-dkim-delegate] Kucherawy, M. and d. dcrocker, "Delegating DKIM Signing Authority", draft-kucherawy-dkim-delegate-02 (work in progress), April 2015.

[I-D.kucherawy-original-authres] Chew, M. and M. Kucherawy, "Original-Authentication-Results Header Field", draft-kucherawy-original-authres-00 (work in progress), February 2012.

[RFC1034] Mockapetris, P., "Domain names - concepts and facilities", STD 13, RFC 1034, DOI 10.17487/RFC1034, November 1987, <http://www.rfc-editor.org/info/rfc1034>.

[RFC1035] Mockapetris, P., "Domain names - implementation and specification", STD 13, RFC 1035, DOI 10.17487/RFC1035, November 1987, <http://www.rfc-editor.org/info/rfc1035>.

[RFC1930] Hawkinson, J. and T. Bates, "Guidelines for creation, selection, and registration of an Autonomous System (AS)", BCP 6, RFC 1930, DOI 10.17487/RFC1930, March 1996,

<http://www.rfc-editor.org/info/rfc1930>.

[RFC4686] Fenton, J., "Analysis of Threats Motivating DomainKeys Identified Mail (DKIM)", RFC 4686, DOI 10.17487/RFC4686, September 2006, <http://www.rfc-editor.org/info/rfc4686>.

[RFC4954] Siemborski, R., Ed. and A. Melnikov, Ed., "SMTP Service Extension for Authentication", RFC 4954, DOI 10.17487/RFC4954, July 2007, <http://www.rfc-editor.org/info/rfc4954>.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, DOI 10.17487/RFC5226, May 2008, <http://www.rfc-editor.org/info/rfc5226>.

[RFC5518] Hoffman, P., Levine, J., and A. Hathcock, "Vouch By Reference", RFC 5518, DOI 10.17487/RFC5518, April 2009, <http://www.rfc-editor.org/info/rfc5518>.

[RFC5863] Hansen, T., Siegel, E., Hallam-Baker, P., and D. Crocker, "DomainKeys Identified Mail (DKIM) Development, Deployment, and Operations", RFC 5863, DOI 10.17487/RFC5863, May 2010, <http://www.rfc-editor.org/info/rfc5863>.

[RFC6377] Kucherawy, M., "DomainKeys Identified Mail (DKIM) and Mailing Lists", BCP 167, RFC 6377, DOI 10.17487/RFC6377, September 2011, <http://www.rfc-editor.org/info/rfc6377>.

[RFC6541] Kucherawy, M., "DomainKeys Identified Mail (DKIM) Authorized Third-Party Signatures", RFC 6541, DOI 10.17487/RFC6541, February 2012, <http://www.rfc-editor.org/info/rfc6541>.

[RFC6672] Rose, S. and W. Wijngaards, "DNAME Redirection in the DNS", RFC 6672, DOI 10.17487/RFC6672, June 2012, <http://www.rfc-editor.org/info/rfc6672>.

[RFC6698] Hoffman, P. and J. Schlyter, "The DNS-Based Authentication of Named Entities (DANE) Transport Layer Security (TLS) Protocol: TLSA", RFC 6698, DOI 10.17487/RFC6698, August 2012, <http://www.rfc-editor.org/info/rfc6698>.

[RFC7208] Kitterman, S., "Sender Policy Framework (SPF) for Authorizing Use of Domains in Email, Version 1", RFC 7208, DOI 10.17487/RFC7208, April 2014, <http://www.rfc-editor.org/info/rfc7208>.

Appendix A. DNS Example of TPA-Label Resource Record placement

```
####
# Practices for Example.com email domain using example.com, isp.com,
# and example.com.isp.com as signing domains.
####

#### 5322.From authorization for 3P domains ####

## "isp.com" TPA-Label Resource Record ##
_HTIE4SWL3L7G4TKAFAUA7UYJSS2BTEOV._smtp._tpa.example.com. IN TXT
  "v=tpa1; tpa=isp.com; param=d;"

#### 5322.Sender/List-ID authorization for 3P domains ####

## "example.com.isp.com" TPA-Label Resource Record ##
_6MEHLQLKWAL5HQREXWDN2TBXAJ6VZ44B._smtp._tpa.example.com.  IN TXT
  "v=tpa1 tpa=*.isp.com; param=d L S;"
```

Appendix B. C code for label generation

The following utility can be compiled as TPA-Label.c using the following:

gcc -lcrypto TPA-Label.c -o TPA-Label

```
<CODE BEGINS>
/*
 * TPA-Label generation utility
 * Copyright (c) 2010 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.
 *
 * This document and the information contained herein are provided on an
 * "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS
 * OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND
 * THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS
 * OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF
 * THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED
 * WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
 */
#include <stdio.h>
#include <sys/types.h>
#include <stddef.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <ctype.h>
#include <unistd.h>
#include <fcntl.h>
#include <errno.h>
#include <openssl/sha.h>

#define TPA_LABEL_VERSION   102
#define MAX_DOMAIN_NAME     256
#define MAX_FILE_NAME       1024

static char base32[] = "ABCDEFGHIJKLMNOPQRSTUVWXYZ234567";
```

```
static char sign_on[] =
{"%s v%d.%02d Copyright (C) (2014)  The IETF Trust\n"};
char err_cmd[] =\
 "ERR: Command error with [%s]\n";
char use_txt[]=\
 "Usage: TPA-Label [-i domain_input_file] [-o label_output_file][-v]\n";
char help_txt[]=\
 "The options are as follows:\n"\
 "-i  domain name input. Defaults to stdin. Removes trailing '.'\n"\
 "-o  TPA-Label output.  Defaults to stdout.\n"\
 "-v  Specifies Verbose Mode.\n\n";

static void usage(void);
/*- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - */

static void
usage(void)
{
    (void) fprintf(stderr, "\n%s%s", use_txt, help_txt);
    exit(1);
}
/*- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - */

int
main (int argc, char * argv[])
{
    int  ret_val, in_mode, out_mode, verbose, done, i, j, k;
    char ch;
    unsigned int len;
    unsigned long b_5;
    char in_fn[MAX_FILE_NAME], out_fn[MAX_FILE_NAME];
    unsigned char in_buf[MAX_DOMAIN_NAME + 2];
    unsigned char sha_res[20], tpa_label[33];
    FILE *in_file, *out_file;

    ret_val = in_mode = out_mode = verbose = done = 0;
    len = 0;

    while ((ch = getopt(argc, argv, "i:o:v")) != -1)
    {
        switch (ch)
        {
            case 'i':
                in_mode = 1;          /* input from file */
                (void) strncpy(in_fn, optarg, sizeof(in_fn));
                in_fn[sizeof(in_fn) - 1] = '\0';
                break;
            case 'o':
```

```
                out_mode = 1;         /* out to file */
                (void) strncpy(out_fn, optarg, sizeof(out_fn));
                out_fn[sizeof(out_fn) - 1] = '\0';
                break;
            case 'v':
                verbose = 1;
                break;
            case '?':
            default:
                (void) usage();
                break;
        }
    };

    if (in_mode)
    {
        if ((in_file = fopen(in_fn, "r")) == NULL)
        {
            (void) fprintf(stderr,
                           "ERR: Error opening [%s] input file.\n",
                           in_fn);
            exit(2);
        }
    }
    else
    {
        in_file = stdin;
    }

    if (out_mode)
    {
        if ((out_file = fopen(out_fn, "w")) == NULL)
        {
            (void) fprintf(stderr,
                           "ERR: Error opening [%s] output file.\n",
                           out_fn);
            exit(3);
        }
    }
    else
    {
        out_file = stdout;
    }

    if (out_mode && verbose)
    {
        (void) printf(sign_on, "TPA-Label utility",
                      TPA_LABEL_VERSION / 100,
```

```
                    TPA_LABEL_VERSION % 100);
    }

    for (i = 0; i < MAX_DOMAIN_NAME && !done; i++)
    {
        if ((ch = fgetc(in_file)) == EOF)
        {
            ch = 0;
        }
        else  if (ch == '\n' || ch == '\r')
        {
            ch = 0;
        }

        in_buf[i] = tolower(ch);

        if (ch == 0)
        {
            len = i;         /* string length */
            done = 1;
        }
    }

    if (!done)
    {
        (void) fprintf(stderr, "ERR: Domain name too long.\n");
        exit (4);
    }

    if (len && in_buf[len - 1] == '.')    /* remove any trailing "." */
    {
        len--;
        in_buf[len] = 0;     /* replace trailing "." with 0 */
    }

    in_buf[len] = 0;         /* terminate string */

    if (len < 2)
    {
        (void)
        fprintf(stderr,
                "ERR: Domain name [%s] too short with %d length.\n",
                in_buf,
                len);
        exit (5);
    }

    SHA1(in_buf, len, sha_res);
```

```
    if (verbose)
    {
        printf("Normalized Domain = [%s] %d, SHA-1 = ", in_buf, len);

        for (i = 0; i < 20; i++)
        {
            printf("%02x", sha_res[i]);
        }
        printf("\nTPA-Label 5 bit intervals left to right.\n");
    }

    /* process sha1 results 4 times by 40 bits (160 bits) */
    for (i = 0, j = 0; i < 4 ; i++)
    {
        b_5 =  (unsigned long long) sha_res[(i * 5)] << 32;
        b_5 |= (unsigned long long) sha_res[(i * 5) + 1] << 24;
        b_5 |= (unsigned long long) sha_res[(i * 5) + 2] << 16;
        b_5 |= (unsigned long long) sha_res[(i * 5) + 3] << 8;
        b_5 |= (unsigned long long) sha_res[(i * 5) + 4];

        if (verbose)
        {
            printf(" {%010llX}->", b_5);
        }

        for (k = 35; k >= 0; k-= 5, j++)    /* convert 40 bits (5x8) */
        {
            tpa_label[j] = base32[(b_5 >> k) & 0x1F];

            if (verbose)
            {
                printf(" %02X:%c",
                       (unsigned int)(b_5 >> k) & 0x1F,
                       tpa_label[j]);
            }
        }
        if (verbose)
        {
            printf ("\n");
        }
    }
    if (verbose)
    {
        printf("\n");
    }
    tpa_label[j] = 0;   /* terminate label string */
    fprintf(out_file, "_%s", tpa_label);
    printf("\n");
```

```
    /* close */
    if (out_mode)
    {
        if (fclose (out_file) != 0)
        {
            (void) fprintf(stderr,
                           "ERR: Unable to close %s output file.\n",
                           out_fn);
            ret_val = 6;
        }
    }
    if (in_mode)
    {
        if (fclose (in_file) != 0)
        {
            (void) fprintf(stderr,
                           "ERR: Unable to close %s input file.\n",
                           in_fn);
            ret_val = 7;
        }
    }
    return (ret_val);
}
<CODE ENDS>
```

Appendix C. History of Prior Efforts

To withstand asserting strict alignment practices, a scheme was devised that transferred the burden of a resulting disruption from receivers back to the Trusted Domains making the stringent requests. As such, a method to authorize other validated domains to establish informally Federated Third-Party Services, such as mailing-lists was developed. This initial scheme was then modified and proposed by ATPS [RFC6541]. Unlike the initial scheme, ATPS required Third-Party Services to use specific non-standard DKIM signatures to signal use of the ATPS authorization strategy. ATPS also required the DKIM signatures used by Third-Party Services to somehow determine the different label encoding employed by the many Trusted Domains without any defined discovery or exchange method.

Both of these changes made deployment impractical by impacting systems not benefiting from additional alignment requirements. Third-parties have often been offering free services for decades. Even renaming From headers would impair normal handling. Those offering these services should not be expected to carry the burden of enabling a new Trusted Domain compliance scheme. Trusted Domains should offer the information needed to avoid disrupting these

services instead, which is the purpose of TPA-Labels.

Rather, the Trusted Domain seeking cooperative handling and receiving receiver feedback necessary to mitigate disruption should handle this burden instead. It is the Trusted Domain that directly benefits after all. There should not be unnecessary and problematic encoding schemes or assertions of delivery chains being expected of any Third-Party Service. Such matters are simply not their concern nor in their benefit.

It seems the added complexities found in ATPS were to defend against a single DNS transaction. However, before this transaction occurs, the Third-Party must permit the validation of their own domain. Even then, a Third-Party checking transaction only occur after the domain is not within the Trusted Domain's alignment assertions. An assertion that can always be removed at any point. It is clearly in the interest of the Trusted Domain where the checking transaction represents a very minor contribution in support of desired receiver cooperation.

Tailoring their TPA-Label list to suit their own users should discourage non-cooperative references to their domain. As more domains reference a common "_tpa." zone, the clout of that zone increases at a very moderate cost. This additional clout better ensures timely responses to abuse notifications. In this manner, DMARC/TPA-Labels would be helping to improve anti-abuse cooperation. In that light, TPA-Labels should be considered a sound investment and not an unwanted burden.

ATPS required new tags be included in Third-Party DKIM signatures. These were "atps" and "atpsh" to construct a chain of "d=" and "atps=". This added complexity without any immediate benefit. Determining optional label encoding without any defined discovery method overlooks that authorization is only possible after the Third-Party Domain has been validated. A complete lack of ATPS deployment should have been expected since necessary changes did not align with benefits.

In contrast, TPA-Labels do not require ANY change be made by authorized third-parties. Disrupting legitimate communications imposes inordinate support costs as a result of erroneously asserting strict alignment practices. The resulting disruption will eventually cause the domain's assertions to be ignored. If this disruption becomes endemic, assertions of other domains will become ignored as well. Domains wishing to benefit from their handling advice being employed while sending legitimate messages that may not retain their asserted alignment practices, should be offering the needed TPA-Label exception information. This information is essential and is only

known by the Trusted Domain through their DMARC feedback.

At this time, it is not practical for large ISPs to make strict DMARC assertions. Strict alignment assertions exclude normal Third-Party Services that modify the requisite alignment. TPA-Label lists specifically tailored to handle their users' desired Third-Party Services will permit their users to have normal email use. While entailing some administrative effort, TPA-Labels will generally be of benefit to their users by widely discouraging any spoofing of their messages. This affords greater protection for the users and the user's recipients.

SPF purported to provide an anti-spoofing feature for an unseen parameter. Nevertheless, its strict IP address authorization causes problems and is largely disregarded for anything other than limiting the sending of DSNs or the scoring of messages. Many institutions will benefit by ensuring their strict DMARC assertions are not disruptive. Exercising this care will help retain recipient's trust in their assertions and the veracity of their messages. TPA-Labels would allow these institutions a means to use informal Third-Party Services with minimal administrative effort. Rather than using subdomains that lack DMARC restrictions, suitable Third-Party Services can be authorized by TPA-Labels. This approach offers a proactive method for recipients to better filter possible phishing attempts by not exposing them to unrestricted subdomain abuse.

TPA-Label publishing is similar to VBR ([RFC5518]). However, it leverages Third-Party validation confirmed by labels held in the Trusted Domain. DNS also permits the information to be transparently made available from other domains whenever desired. TPA-Labels provide domains a means to protect their recipients while still permitting the use of legitimate SMTP exchanges. By implementing DMARC/TPA-Labels, these domains should be better able to stand on their own merit.

Author's Address

Douglas Otis
Trend Micro
10101 N. De Anza Blvd
Cupertino, CA 95014
USA

Phone: +1.408.257-1500
Email: doug_otis@trendmicro.com