Sender Policy Framework: Email Address Internationalization
draft-ellermann-spf-eai-06

Abstract

UTF8SMTP is an extension of SMTP (Simple Mail Transfer Protocol) allowing the use of UTF-8 in the SMTP envelope for EAI (Email Address Internationalization) and message headers. This memo discusses the consequences for SPF (Sender Policy Framework).

Editorial note

This note, the IANA considerations (Section 6), and the document history (Appendix A) should be removed before publication. For some imported terms in Section 1.1 the sources have to be fixed. The draft can be discussed on the <mailto:spf-discuss@listbox.com> mailing list.

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 May 15, 2012.

Copyright Notice

Copyright (c) 2011 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
# Table of Contents

1. Introduction .............................................. 4  
   1.1. Imported terms ................................... 4  

2. Background .............................................. 4  
   2.1. SPF HELO identity ................................. 4  
   2.2. EAI MAIL FROM identity ......................... 5  
   2.3. Local parts ...................................... 6  

3. Details .................................................. 7  
   3.1. Considerations for SPF publishers ............. 8  
   3.2. Received-SPF ..................................... 8  
   3.3. Internationalization of explanations ........... 9  

4. Acknowledgements ........................................ 10  

5. Security Considerations ............................... 10  

6. IANA Considerations .................................... 11  

7. References .............................................. 11  
   7.1. Normative References ............................ 11  
   7.2. Informative References .......................... 12  

Appendix A. Document History ............................. 12  

Author’s Address ......................................... 14
1. Introduction

Readers should be familiar with SMTP as specified in [RFC5321] and the SPF terminology in [I-D.kitterman-4408bis].

The keywords "MUST NOT" and "SHOULD" in this memo are to be interpreted as described in [RFC2119].

For an EAI (Email Address Internationalization) overview see [RFC4952]. UTF-8 is specified in [RFC3629]. An MTA is a Mail Transfer Agent, e.g., an SMTP relay.

1.1. Imported terms

Some [RFC5234] ABNF terms used in this memo are specified in other sources:

- `<domain-spec>` = <see 4408bis section 8.1>
- `<target-name>` = <see 4408bis section 4.8>
- `<explanation>` = <see 4408bis section 6.2>
- `<textstring>` = <see RFC5321 section 4.2>
- `<Dot-string> = <see RFC5321 section 4.1.2>
- `<Quoted-string>` = <see RFC5321 section 4.1.2>
- `<Atom>` = <see RFC5321 section 4.1.2>
- `<atext>` = <see RFC5322 section 3.2.3>
- `<UTF8-non-ascii>` = <see RFC5336 section 2.3>
- `<uMailbox>` = <see RFC5336 section 2.3>

2. Background

SMTP as specified in [RFC5321] supports only ASCII addresses and LDH (letter, digit, hyphen) domain labels. The letters are ASCII letters; certain LDH-labels are also known as A-labels in the context of IDN (Internationalization of Domain Names) and [RFC5890].

2.1. SPF HELO identity

In SMTP sessions after an SMTP EHLO command from the client the server response can indicate supported SMTP extensions. [RFC5336] specifies the UTF8SMTP extension.

The SMTP client can accept an offered UTF8SMTP extension by using one of the specified features, notably by the use of UTF-8 in mailbox addresses of SMTP commands, by the use of alternative ASCII addresses in these commands, or by the use of UTF-8 in the message header for addresses and other purposes, i.e. by sending a "message/global" instead of a "message/rfc822" as specified in [RFC5335].
Because UTF8SMTP support is indicated in the response to an EHLO command it cannot be used after HELO, and the SPF HELO identity is not affected by EAI: The domain in a HELO or EHLO command consists of ordinary LDH-labels, or it is a domain literal. For an empty reverse path, as it is used in non-delivery reports and other auto-replies, SPF fabricates a MAIL FROM identity based on the HELO identity with a case insensitive local part "postmaster"; this scenario is also not affected by EAI.

A domain consisting of LDH-labels including IDN A-labels beginning with "xn--" is an ordinary LDH-domain as far as DNS (Domain Name System), SPF, and UTF8SMTP are concerned. Apart from HELO and EHLO the only relevant SMTP command for SPF is the MAIL FROM command with the reverse path containing the envelope sender address (if it is not empty, see above). When the derived MAIL FROM identity is an ordinary address SPF can handle it as specified in [I-D.kitterman-4408bis].

2.2. EAI MAIL FROM identity

The interesting UTF8SMTP cases for SPF contain non-ASCII UTF-8 characters in the local part (left hand side) or the domain part (right hand side) of the MAIL FROM identity. Domain labels containing non-ASCII UTF-8 characters are also known as U-labels in IDN.

SPF checks typically make only sense at the "border MTA", and this is normally an MX (Mail eXchanger) of the receiver talking with a sender. An MTA wishing to check the SPF sender policy against the IP of the sender fetches the sender policy for the domain in the HELO or MAIL FROM identity as DNS client of a server for the alleged sender. The SPF terminology might be confusing: The border MTA is the SMTP server, but for the purpose of checking a sender policy it is the SPF (or rather DNS) client of a name server for the alleged sender with a given HELO or MAIL FROM identity.

An MTA could "downgrade" EAI MAIL FROM addresses using an optional alternative address given as UTF8SMTP MAIL-parameter. Where that happens the resulting new MAIL FROM address is an ordinary [RFC5321] reverse path and can be handled as usual.

Skipping all ordinary cases as noted above the SPF client confronted with an EAI address in the MAIL FROM identity is generally an MTA supporting UTF8SMTP, and supposed to know how to transform U-labels into corresponding A-labels, e.g., because it might need to send a non-delivery report to the envelope sender address later; see [RFC5337].
For agents trying post-SMTP SPF-checks this might be not the case, and unsurprisingly attempts to fetch the sender policy of a domain with U-labels "as is" will fail with SPF result NONE. Arguably this is a broken setup, the border MTA should not offer and accept UTF8SMTP mails if critical parts behind it - not limited to the mailbox of the receiver - don’t support EAI.

2.3. Local parts

Top down at this point the remaining SPF clients are supposed to know how to transform U-labels into A-labels, and fetch the SPF policy of the alleged sender. SPF implementors and publishers of SPF sender policies should note that only the domain part of the MAIL FROM identity is transformed from U-labels into A-labels. The local part MUST NOT be transformed, it is used "as is" in the construction of a <target-name> by SPF macro expansion involving local part macros.

SPF allows all octets in labels of a <target-name> excluding dots, which are supposed to separate labels. Sender policies can directly talk about any <domain-spec> with labels separated by dots, where each label consists of 1 to 63 visible ASCII characters except "%" introducing macros. The macros "%%" and "%_" in a <domain-spec> expand into "%" or space in the <target-name>, respectively.

Normally sender policies do not use such slightly odd labels, and the most extravagant case is "_" (underscore) in a label that is intentionally no LDH-label. Nevertheless implementations have to support such oddities, because they are needed in the case of a <target-name> derived from a <domain-spec> using the local part macro.

SMTP local parts can have two forms, <Dot-string> or <Quoted-string>. A <Dot-string> consists of dot separated <Atom>s, each <Atom> consists of one or more <atext> characters. Please note that an <Atom> is not the same as an LDH-label, it is also not the same as a domain label, e.g., <Atom>s can be longer than 63 octets.

By definition there are no leading, trailing, or adjacent dots in a <Dot-string>. [RFC5321] and its predecessor recommend to avoid the <Quoted-string> form of a local part. Current SPF implementations are known to strip the quotes from a <Quoted-string> for the purpose of determining a <target-name> derived from a <domain-spec> using the local part macro. This can result in an invalid <target-name> with leading, trailing, or adjacent dots, e.g., for a mail address "do..ts@example.org.

Publishers of sender policies using the local part macro SHOULD make sure that the used pieces of valid local parts in their domain can be
parsed into non-empty domain labels; one way to achieve this is to avoid <Quoted-string>. The effect of a <Quoted-string> local part is not clearly specified in [I-D.kitterman-4408bis]. In theory DNS supports any octet, even "embedded" dots within a label. In practice current SPF implementations cannot handle embedded dots, and it is far from clear that quoted pairs introduced by a "\" (backslash) in a <Quoted-string> are interpreted as specified in [RFC5321] section 4.1.2.

Publishers of sender policies using the local part macro SHOULD make sure that the used pieces of valid local parts in their domain result in 1 to 63 octets per dot separated domain label as mentioned in [I-D.kitterman-4408bis] section 8.1. Please note that the truncation of longer labels after macro expansion is not clearly specified: SPF implementations could truncate longer labels left to right or right to left, they could also ignore affected directives, or treat this case as error.

Publishers of sender policies using the local part macro need to be aware that ASCII letters in the used pieces of valid local parts in their domain are in essence treated as case-insensitive by DNS as explained in [RFC4343].

UTF8SMTP extends <atext> by <UTF8-non-ascii>, and it also permits <UTF8-non-ascii> in quoted strings. As far as SPF is concerned <UTF8-non-ascii> can result in non-ASCII octets in a <target-name>, working ”as is” in DNS labels with similar caveats as noted above with respect to the length of labels, case sensitivity, and normalization.

[RFC5335] suggests to restrict the use of UTF-8 in EAI addresses to Normalization Form C (NFC) as recommended in [RFC5198]. Publishers of sender policies using the local part macro need to be aware that SPF implementations treat local parts "as is". Mapping different forms of an EAI local part to one mailbox at their border MTAs has no effect on different forms of EAI local parts in DNS queries. A straight forward strategy to avoid potential issues with respect to SPF is to use local part macros only in non-critical explanations and maybe for logging, if at all.

3. Details

For historical reasons technical SPF [Errata] have been "outsourced". SPF implementors are hopefully also interested in the SPF test suite on the same site.
3.1. Considerations for SPF publishers

Policy publishers should know that this memo does not update [I-D.kitterman-4408bis], in theory EAI is compatible with SPF. It is not possible to use U-labels in sender policies directly, they have to be transformed into the corresponding A-labels. Likewise U-labels in UTF8SMTP MAIL FROM addresses are transformed into A-labels for the purposes of SPF by implementations supporting EAI.

SPF implementations not supporting U-labels in MAIL FROM identities will return NONE instead of the intended result, e.g., PASS or FAIL. UTF8SMTP senders wishing to avoid this problem could transform MAIL FROM U-labels into A-labels on their side. They could also hope that spammers forging MAIL FROM identities will not abuse IDN U-labels in the near future, and that most SPF implementations will be updated before this changes. Unfortunately experience has shown that spammers learn faster than lazy users.

MAIL FROM identities using only A-labels, with or without UTF-8 in the local part, work "as is" for the purposes of SPF. HELO identities consist either of A-labels, are domain literals and irrelevant for SPF, or are syntactically malformed as far as UTF8SMTP and SPF are concerned. SPF does not specify how receivers should handle SMTP syntax errors.

UTF8SMTP allows to specify an optional alternative address in the traditional syntax. Receivers are free to check SPF also or only based on the alternative address. Obviously a sender policy for the alternative address should permit the same sending IPs as the sender policy for the EAI address, and one simple way to achieve this is to use corresponding A-labels in an alternative address yielding one SPF sender policy for both addresses. Please note that this is not required by UTF8SMTP, it permits to use unrelated domains with different policies. Clearly if some IPs permitted for one address fail for the other address, or vice versa, the sender will have problems, if the affected IPs are actually used to send mails.

3.2. Received-SPF

The Received-SPF header field specified in [I-D.kitterman-4408bis] section 7 is a "message/rfc822" trace header field.

UTF8SMTP transports a "message/global" as specified in [RFC5335] permitting the use of UTF-8 in header fields. For Received-SPF this is necessary to record an UTF8SMTP "envelope-from" <uMailbox> address.

UTF-8 might be also needed in comments and other parts of this header.
field in conjunction with UTF8SMTP. See [RFC5335] for the corresponding syntax modifications.

SPF implementations could check the EAI MAIL FROM *and* an alternative address (if given). In this case SPF implementations SHOULD record both results. An exception could be to omit a "less interesting" (e.g., equivalent, NONE, or NEUTRAL) result. Receivers could also adopt the strategy to check the second of two addresses only if the result of their first check is "not helpful" (e.g., NONE or NEUTRAL).

3.3. Internationalization of explanations

SMTP replies consist of a reply code and optionally a <textstring> as explanation. This is the vehicle used by SMTP servers if they reject a mail after an SPF FAIL, optionally adding an explanation given in the SPF policy as <explanation>.

The <textstring> is limited to ASCII and hopefully integrated in some way in any resulting non-delivery report (bounce) created by the SMTP client. UTF8SMTP does *not* modify this restriction to ASCII <textstring>s in contexts relevant for SPF.

The optional SPF <explanation> is in essence a pointer to a domain with a TXT record. The macro-expanded content of this TXT record can be used in the SMTP reply.

The resulting explanation can contain a URL of a Web page with internationalized explanations including data based on the sending IP, sender address, and other details given as SPF macros in the TXT record.

The upper-case forms of SPF macros trigger URL-encoding for this purpose. SPF macro expansion does not involve to parse textual explanations for potential URLs, it percent-encodes any "reserved" input character. For EAI these characters can be UTF-8 and have to be percent-encoded as specified in [RFC3987]. This boils down to "take octets as is, if not unreserved percent-encode".

The procedure outlined above clearly cannot guarantee to produce valid URLs in an explanation. In fact any use of lower-case macros could result in non-ASCII explanations, and in that case the SMTP server cannot use the result in the <textstring>s of its reply after an SPF FAIL.
4. Acknowledgements

Various folks on the SPF discuss and devel mailing lists worked on the SPF [Errata] and test suite. All obscure issues with local parts discussed in this memo are based on their prior work and indirectly related discussions on the EAI and SMTP mailing lists.

Thanks to Abel Yang, Alessandro Vesely, Boyd Lynn Gerber, Doug Otis, Harald T. Alvestrand, John C. Klensin, Julian Mehnle, Kari Hurtta, Kazunori Fujiwara, Scott Kitterman, Stuart D. Gathman, and Wayne Schlitt for their encouragement, feedback, or critique.

5. Security Considerations

When UTF8SMTP senders use a different domain in the optional alternative MAIL FROM address, and the corresponding sender policy is also different, it is hard to predict which policy will be checked, if any, depending on the route to the receiver and other factors. Different results can be hard to diagnose, e.g., if a mail from the same sender to the same receiver sometimes results in PASS and at other times in FAIL. One proposal to mitigate this problem is discussed in Section 3.1.

Not all SPF implementations will already support U-labels as explained in this memo. Senders could transform U-labels in MAIL FROM commands to A-labels on their side where this is a problem.

Using the SPF local part macro in conjunction with EAI is not intuitive, local parts are not transformed to A-labels. This is not new, but in conjunction with EAI very likely. Various details with respect to label lengths, quoted strings, adjacent dots, and quoted pairs are explained in Section 2.3. Not using <Quoted-string>s as local parts and not using case-sensitive local parts is recommended in [RFC5321] section 4.1.2; this also covers quoted pairs and adjacent dots.

A new UTF8SMTP problem is the use of local part macros for the construction of "per user" policies, when different variations of an UTF-8 local part correspond to one user mailbox. While the use of normalized UTF-8 strings is recommended in [RFC5335] section 4.1 this does not help with case-sensitive <UTF8-non-ascii> variations. One way to address this problem is to avoid critical uses of the local part macro as discussed in Section 2.3.

UTF8SMTP servers can be forced to send non-delivery reports to forged envelope sender addresses, if some receiver mailboxes can handle EAI mails, while others can’t, and the server has no way to "downgrade"
mails to traditional receivers. Hopefully a future SMTP extension will allow a kind of "selective reject" mechanism. Publishing SPF PASS or FAIL policies, and rejecting FAIL at the border MTA, would eliminate this problem. Similarly non-delivery reports after a PASS cannot hit innocent bystanders.

Evaluating PRA (Purported Responsible Address) policies as specified in [RFC4406] with SPF or vice versa can cause havoc, as the algorithms are semantically different even when the policies are otherwise syntactically identical. This known problem is discussed in [I-D.kitterman-4408bis].

SPF local part macros in a <domain-spec> could be abused in an attack scenario to avoid DNS caching. Compare the SPF processing limits in [I-D.kitterman-4408bis] section 10.1.

6. IANA Considerations

Keep up the good work, nothing to do here.

7. References

7.1. Normative References


[RFC5335] Abel, Y., "Internationalized Email Headers", RFC 5335,
September 2008.


7.2. Informative References


Appendix A. Document History

Changes in version 06:

* Fixed broken inline references by using the xml2rfc release version instead of the experimental version.

* Replaced the openspf.org link for the SPF test suite and [Errata] by openspf.net temporarily.
Changes in version 05 (2011):

* Update for UTF8SMTP while UTF8SMTPBIS is not yet approved. Only
  the references and boilerplate texts were updated; otherwise the
  content is as it was in 2008.

* The IESG put the "outsourced" [Errata] on hold for an SPF document
  update; the errata are covered by [I-D.kitterman-4408bis].

* Meanwhile [RFC5321], [RFC5322], [RFC5451], [RFC5890], and
  [I-D.kitterman-4408bis] exist.

Changes in version 04 (2008):

* Fixed two typos and the references for the published EAI RFCs.

Changes in version 03:

* Added a terminology Section 1.1. Apart from the added [RFC5322]
  defining <atext> all referenced drafts are now approved RFCs.

* Clarified that case-insensitive uses of <UTF8-non-ascii> can cause
  havoc in conjunction with local part macros. This is not
  mitigated by the recommended normalization.

* Added a pointer to the [I-D.kitterman-4408bis] security
  considerations in Section 5. Mentioned the role of local part
  macros in an attack scenario with credits to Doug Otis.

* Added the missing credits for the SPF discussions on the EAI list
  back in 2006 and 2007.

* Added a link to the "outsourced" [Errata] mentioning the SPF test
  suite. Errata and test suite cover some <target-name> issues
  discussed in this memo, notably "adjacent dots".

* Added a long Section 3.3 about the wonders of SPF explanations
  with the rather trivial conclusion that [RFC3987] percent-encoding
  works as expected.

Changes in version 02:

* This memo is about EAI, not wildcard issues elsewhere; replaced
  4592 by [RFC4952].

* Noted that only certain LDH-labels are or might be A-labels. The
  details are specified in [RFC5890].
* [RFC5451] not yet added, extending the Authentication-Results header field to allow UTF-8 values for EAI should be straightforward.

Changes in version 01:

* Extended the local part discussion in Section 2.3 significantly to cover known gaps in the SPF specification. Added potential issues with non-normalized local parts.

* Added a Received-SPF section using the now "Last Called" [RFC5335]. Added references to [RFC4343], [RFC4952], [RFC5198], and the approved [RFC5337].

* Removed the discussion of two vs. five SPF "subtypes", this belongs into another draft. Kept the caveat about using an algorithm designed for subtype x on a policy with subtype y (or v.v.), as this could cause hard to debug mail losses.

Initial version:

* In a short discussion on the EAI list Harald Alvestrand and John Klensin encouraged to collect SPF EAI considerations in a separate memo.

Author’s Address

Frank Ellermann
xyzzy
Hamburg, Germany

Email: hmdmhdfhdjzmzdjtmzdtktdkdjz@gmail.com
URI: http://purl.net/xyzzy/