

EnvelopeSenderInReceived

A Proposal for Storing Envelope-Sender Data in Received Headers

Traditionally, the envelope-sender data has been added to messages as they pass through MTAs, but in varying ways (the Return-Path header, X-Envelope-From, etc.), and generally in terms of one header in the message. see [EnvelopeSenderInHeaders](#).

A very helpful addition to support MUAs and MUA-level filters, would be if this was added to the Received header for the MTA handover, e.g.

```
Received: from portent.example.com (portent.example.com [208.58.1.195])
  by bar.example.org (envelope-from
  <owner-foo-discuss@example.com>)
  (Postfix) with ESMTTP id 12CE931024F for <foo@example.org>;
  Wed, 28 Jan 2004 08:54:19 +0000 (GMT)
```

This would allow filters like [SpamAssassin](#) to pick up the envelope-from used at each step of the chain, which is very valuable especially when intermediate steps tend to rewrite it. (For example, fetchmail makes some incorrect assumptions, and will add an *incorrect* Return-Path header if an X-Envelope-From header exists, even from an earlier handover.)

If I recall correctly, some versions of exim seem to support this.

We already have support for this in [SpamAssassin](#) – we detect it using this regexp matched against Received lines:

```
/envelope-(?:sender|from)[ =](\S+)\b/
```

MTA authors/packagegers, please consider adding this to your default configurations; inserting a string in the form

```
(envelope-sender <foo@example.com>)
```

somewhere after the "from ... by ..." text would do the trick nicely.

MTAs That Do This

If you know of an MTA package that will add the envelope-sender data to the Received header, or can write a configuration stanza to add to config files to do this in any given MTA, please add a note about it on this page using the "EditText" button below.

- [QmailSpfPatch](#) added by Ed Li <zixia@zixia.net>
Qmail always save envelope-from in Return-Path header, but if: 1. We must forward mail to another server, and use [SpamAssassin](#) in that server (Return-Path may be changed); 2. any poor guy like me: can't make Return-Path work with SPF(with [SpamAssassin](#) 3.0.0-r20550); We need this patch 😊.
- Exim 4
Exim 4 can be configured to output the envelope-sender by adding a custom received_header_text option. The following adds the envelope-sender to the default received header template:

```
received_header_text = Received: \
  ${if def:sender_rcvhost {from $sender_rcvhost\n\t}\
  ${if def:sender_ident {from $sender_ident }}\
  ${if def:sender_helo_name {(helo=$sender_helo_name)\n\t}} }\
  by $primary_hostname \
  ${if def:sender_address {(envelope-from\n\t\
  <$sender_address>)\n\t}}\
  ${if def:received_protocol {with $received_protocol}} \
  ${if def:tls_cipher {(tls_cipher)\n\t}}\
  (Exim $version_number)\n\t\
  id $message_id\
  ${if def:received_for {\n\tfor $received_for}}
```

- Sendmail
Mike Markley reports: Sendmail can be trivially made to insert the envelope-sender into the Received header by setting confRECEIVED_HEADER, or by using the builtin *REC_BY* macro (used to construct the default Received header) like:

```
define(`_REC_BY_', `$.by $j (envelope-from $f) ($v/$Z)$?r with $r$. id $i${tls_version}')
```

Comments

Actually, the Return-Path is standard now, and I don't see any need to add yet another way to specify the envelope recipient. Go read RFC-2821 section 4.4 where this header is defined as a MUST. – Arik Baratz

The issue is that MUAs and MUA-level filters like [SpamAssassin](#) cannot reliably determine the envelope sender (note; not recipient) at *intermediate* steps along the Received chain. Some forwarding use cases, like mailing lists, fetchmail, or SRS-compliant forwarders, will resend the message with a new envelope sender, which means that a filter cannot check back beyond that step. Given situations where spam arrives via lists or at a forwarding address, this means that the filter cannot use the env-sender to detect spam in that case, and in the fetchmail situation, the filter cannot trust that data at all, and therefore cannot use env-sender or SPF at all. Recording Return-Path is still useful and required, but this is an 'extra' which allows this data to be visible for all stages of the message relay flow. – [JustinMason](#)

But [SpamAssassin](#) sometimes runs as an MTA-level filter - [example at the top shows SA refusing mail at SMTP time](#). It should run as such wherever possible, IMO, and when it does, it should reliably determine the envelope. – [MatthewElvey](#)

However, a way for the MTA plugin to *inform* the SA code of the envelope data reliably is needed. The recommended way to do this, currently, is to ensure that the message text passed to SA contains that relay's Received header. Adding the envelope-sender info along with the other envelope info (HELO, connecting IP, rDNS) makes sense here. – [JustinMason](#)

First implemented in SA 3.0.3, in case you were wondering. See bug ID #3944. – Thomas Eisenbarth