

SMTP behavior difference between return-path, reply-to and from

Just random stuff - Nothing to see here - Move along...

SMTP behavior difference between return-path, reply-to and from

Let's start with a simple example. Let's say you have an email list, that is going to send out the following RFC2822 content. (References: [VERP RFC Bounce Messages](#))

```
From: <coolstuff@mymailinglist.com>
To: <you@yourcompany.com>
Subject: Super simple email
Reply-To: <coolstuff-threadId=123@mymailinglist.com>
```

This is a very simple body.

Now, let's say you are going to send it from a mailing list, that implements VERP (or some other bounce tracking mechanism that uses a different return-path). Lets say it will have a return-path of coolstuff-you-yourcompany.com@mymailinglist.com. The SMTP session might look like:

```
{S}220 workstation1 Microsoft ESMTp MAIL Service
{C}HELO workstation1
{S}250 workstation1 Hello [127.0.0.1]
{C}MAIL FROM:<coolstuff-you=yourcompany.com@mymailinglist.com>
{S}250 2.1.0 me@mycompany.com...Sender OK
{C}RCPT TO:<you@yourcompany.com>
{S}250 2.1.5 you@yourcompany.com
{C}DATA
{S}354 Start mail input; end with <CRLF>.<CRLF>
{C}From: <coolstuff@mymailinglist.com>
To: <you@yourcompany.com>
Subject: Super simple email
Reply-To: <coolstuff-threadId=123@mymailinglist.com>

This is a very simple body.
.

{S}250 Queued mail for delivery
{C}QUIT
{S}221 Service closing transmission channel
```

Where {C} and {S} represent Client and Server commands, respectively.

The recipient's mail would look like:

```
Return-Path: coolstuff-you=yourcompany.com@mymailinglist.com
From: <coolstuff@mymailinglist.com>
To: <you@yourcompany.com>
Subject: Super simple email
Reply-To: <coolstuff-threadId=123@mymailinglist.com>

This is a very simple body.
```

Now, let's describe the different "FROM"s.

1. The Return-Path (sometimes called the Reverse-Path or Envelope-FROM — all of these terms can be used interchangeably) is the value used during the SMTP session. As you can see, this does not need to be the same value that is actually found in the mail headers. Only the recipient's mail server is supposed to add a Return-Path header to the top of the email. This records the actual Return-Path sender during the SMTP session. If a Return-Path header is already exists in the email, then that header is to be removed, and replaced by the recipient's mail server. All bounces that occur during the SMTP session should go back to the Return-Path value. Some servers may accept all email, and then queue it locally, until it has a free thread to deliver it to the recipient's mailbox. If the recipient doesn't exist, it should bounce it back to the recorded Return-Path value. Note, not all mail servers obey this rule. Some mail servers will bounce it back to the FROM address.
2. The FROM address is the value actually found in the FROM header. This is supposed to be who the message is FROM. This is what you see as the "FROM" in most mail clients. If an email does not have a Reply-To header, then all human (mail client) replies should go back to the FROM address.
3. The Reply-To header is added by the sender (or the sender's software). It is where all human replies should be addressed too. Basically, when the user clicks "reply", the Reply-To value should be the value used as the recipient of the newly composed email. The Reply-To value should not be used by any server. It is meant for client side use. However, as you can tell, not all mail servers obey the RFC standards or recommendations.

Related Posts

[Some Cool Open Source Software](#)

[Great Open Source Terminal Applications](#)

[DigitalOcean Droplets – Debian Install](#)

EMAIL RFC SMTP

[« Previous: Archiving Gmail](#)

[Next: DigitalOcean Droplets – Debian Install »](#)

RECENTLY SEARCHED FOR...

