

Easy VERP in Exim

By [Roger Keays](#), 1 May 2007

Variable Envelope Return Path (VERP) [1] is a method of sending mail such that the recipients address is encoded into the bounce return path. It is typically used by automatic bounce management software since standard mailing methods don't reliably maintain the original recipient's address.

To support very large mailing lists, you almost certainly want the VERP expansion of addresses to be handled by the MTA. Both the Exim manual and the Mailman HOWTO include a section on VERP expansion [2,3], however the methods described in these documents are more complicated than necessary. Here is a very simple way to do VERP expansion using the Exim MTA.

This method uses a single redirect router to modify the envelope return path. A message with multiple recipients can be passed to this router and, once processed, each rewritten address will reprocessed independently. This means there is no overhead of passing a separate mail to the MTA for each recipient.

```
#
# The VERP router rewrites the address to use for the MAIL FROM command.
#
# Eg: MAIL FROM: bounces@ninthavenue.com.au
#      RCPT TO: test@ilikespam.com
#
# is sent out with
#      MAIL FROM: bounces-test=ilikespam.com@ninthavenue.com.au
#      RCPT TO: test@ilikespam.com
#
verp_out:
    driver = redirect
    condition = ${if match ${local_part:$return_path}}{^bounces\$} {yes}{no}
    errors_to = ${local_part:$return_path} - $local_part = $domain @ ${domain:$return_path}
    data = $local_part @ $domain
```

Test the above configuration with the following command:

```
exim4 -d -f bounces@ninthavenue.com.au -bt test@ilikespam.com
```

More information can be incoded into the bounce address if required. In this example, the software has generated a delivery id for the particular delivery (10), and a recipient-specific VERP number (4) which is used to detect consecutive bounces to that recipient. These values make calculating bounce

statistics more efficient since it removes the requirement that each recipient for every delivery be kept on record.

```
#
# The VERP router rewrites the address to use for the MAIL FROM command;
# and removes the VERP number for the next RCPT TO.
#
# Eg: MAIL FROM: bounces-10@ninthavenue.com.au
#      RCPT TO: 4-test@ilikespam.com
#
# is sent out with
#      MAIL FROM: bounces-10-4-test=ilikespam.com@ninthavenue.com.au
#      RCPT TO: test@ilikespam.com
#
verp_out:
    driver = redirect
    condition = ${if match ${local_part:$return_path}}{^bounces-[^=]+\}$} {yes}{r
errors_to = ${local_part:$return_path}-${local_part=$domain@${domain:$return_p
data = ${sg${local_part}}{^.+?-}{}}@$domain
```

Processing incoming bounces is very simple:

```
# incoming verp router
verp_in:
    driver = accept
    condition = ${if match ${local_part}}{^bounces-} {yes}{no}}
    transport = verp_bounce
```

Although you will, of course, have to implement the verp_bounce transport to invoke your bounce management software.

Using the redirect router for VERP expansion has been working well for me. When compared to using a transport, it is simpler to read and maintain, allows the recipient address to be rewritten as well as the return path and allows you to use your existing set of transports (e.g. local, remote, virtual) without having to add VERP capabilities to each.

References

- [1] <http://cr.yp.to/proto/verp.txt>
- [2] http://www.exim.org/exim-html-4.66/doc/html/spec_html/ch47.html#SECTverp
- [3] <http://www.exim.org/howto/mailman21.html#verpex>

About Roger Keays



Roger Keays is an artist, an engineer, and a student of life. He has no fixed address and has left footprints on 40-something different countries around the world. Roger is addicted to surfing. His other interests are music, psychology, languages, the proper use of semicolons, and finding good food.