Exim Autoreply Example (Exim4)

By Roger Keays, 10 October 2010

Here is some example Exim configuration for email autoreply where the email accounts and response text are kept in a postgresql database. First add a new router to the routers section like this one:

```
#
# Autoreply router which may pass messages to the autoreply transport.
# Unseen must be used to ensure the message is also delivered by passing
# it to the next router.
#
virtual_autoreply:
  driver = accept
  transport = autoreply
  unseen
  condition = ${lookup pgsql{SELECT 1 FROM email_accounts a \
            WHERE a.localpart = '${quote_pgsql:$local_part}' AND \
                  a.domain = '${quote_pgsql:$domain}' AND \
                  a.autoreply = true}}
```

And in the transports section add the following.

```
#
# Autoreply transport. The entire autoreply email needs to built from
# scratch.
#
autoreply:
  driver = autoreply
  to = ${reply_address}
  from = ${local_part}@${domain}
  subject = Re: $h_subject:
  text = ${lookup pgsql{SELECT autoMessage FROM email_accounts a \
            WHERE a.localpart = '${quote_pgsql:$local_part}' AND \
                  a.domain = '${quote_pgsql:$domain}' AND \
                  a.autoreply = true}}
```

There is also a 'once' option on the transport to prevent duplicate autoreplies being sent, but I haven't implemented it in this example because it is unclear from the exim docs if this will work with multiple

email accounts. That is, if more than one user on the server has an email autoreply enabled, we want to reply once per user, not just once total as the docs seem to indicate.

Let me know if I interpreted the documentation wrongly :)

About Roger Keays

Roger Keays is an artist, an engineer, and a student of life. He has no fixed addressand has leftfootprints 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.