

Configurare exim4 per mandare la posta attraverso Gmail

Articolo originale: <http://appsparsi.blogspot.com/2011/01/configurare-exim4-per-mandare-la-posta.html> . Per velocizzare il tutto, allego il file della configurazione per DEBIAN 5

Si dà per scontato che sia già installato exim4: `apt-get install exim4`

Fase 1: configurazione di exim4

- da terminale, digitate `sudo dpkg-reconfigure exim4-config`
- si aprirà un wizard, che vi guiderà passo passo nella configurazione di exim4;
- alla prima scelta che si presenta, scegliamo *posta inviata tramite uno «smarthost»; ricevuta via SMTP o fetchmail*
- nella pagina successiva vi verrà richiesto il nome di dominio della macchina locale, ovvero quella parte di nome che segue la @ in qualsiasi indirizzo locale,
- la domanda successiva riguarda gli indirizzi IP sui quali attendere connessioni SMTP in ingresso: qui dovete rispondere `127.0.0.1`
- nella pagina dopo (Altre destinazioni per conto delle quali accettare posta) potete lasciare il campo in bianco, quindi premete *Invio*
- stessa cosa per il campo successivo (Sistemi per i quali fare il «relay»): lasciate il campo in bianco e premete *Invio*
- subito dopo vi verrà chiesto l'indirizzo IP o hostname per lo «smarthost» in uscita: qui dovete inserire `smtp.gmail.com::587`
- alla domanda successiva (Omettere il mail name locale dai messaggi in uscita?) rispondete NO
- dopo un paio di invii, vi verrà richiesto se mantenere al minimo il numero di richieste DNS (dial-on-demand), anche qui dovete rispondere NO
- alla richiesta di modalità di consegna per la posta locale scegliete *Formato mbox in /var/mail/*
- poi vi chiederà se dividere la configurazione in molti piccoli file, rispondete NO
- l'ultima richiesta (Destinatari della posta di root e postmaster) potete lasciarla in bianco e premere *Invio*
- a questo punto il wizard terminerà e verrà riavviato in automatico il servizio di mail (MTA)

Fase 2

- sempre da terminale con permessi di root, editate il file `/etc/exim4/exim4.conf.template` digitando `gksu gedit/etc/exim4/exim4.conf.template`
- trovate la riga `".ifdef DCconfig_smarthost DCconfig_satellite"`, e in quella sezione (prima della riga `".endif"`) aggiungete le seguenti righe

send_via_gmail:

`driver = manualroute`

`domains = ! +local_domains`

`transport = gmail_smtp`

`route_list = * smtp.gmail.com`

se nel file che state editando trovate qualsiasi altro **smarthost** che contenga la riga `"domains = ! +local_domains"`, dovete cancellare o commentare (con # all'inizio della riga) tutte le righe che lo riguardano,

- trovate la riga `"begin authenticators"`, e aggiungeteci sotto le seguenti righe

```
gmail_login:
driver = plaintext
public_name = LOGIN
client_send = : <indirizzo gmail> : <password>
```

sostituendo i campi "*<indirizzo gmail>*" e "*<password>*" con i vostri dati

- cercate se ci sono altri authenticators che contengono la stessa riga "*public_name = LOGIN*", se esistono cancellateli o commentateli
- cercate il commento:

```
#### transport/30_exim4-config_remote_smtp_smarthost"
```

e aggiungete sotto, in quella sezione

```
gmail_smtp:
    driver = smtp
    port = 587
    hosts_require_auth = $host_address
    hosts_require_tls = $host_address
```

- in fondo al file, esiste una sezione che comincia con questa riga

```
".ifndef AUTH_CLIENT_ALLOW_NOTLS_PASSWORDS"
```

e finisce dopo 31 righe con un ".endif": bene, questa sezione va tutta cancellata o commentata (attenzione che di quelle righe ce ne sono due, partite a commentare dalla prima),

- Salvate e chiudete il file

E ora siamo pronti a passare alla prossima fase

Fase 3

- da terminale, eseguite `sudo update-exim4.conf`
- e infine eseguite `sudo /etc/init.d/exim4 restart`

Se non notate messaggi di errore, vuol dire che tutto è andato bene: ora il vostro **exim4** è pronto per spedire la posta via **Gmail**.

Per provarne velocemente il funzionamento, spedite una mail da riga di comando e poi controllate nella vostra casella **Gmail** se la ricevete:

```
echo "corpo del messaggio " | mail -s "oggetto della mail" <indirizzo gmail>
```

Per fare più in fretta nella configurazione, allego il file di configurazione `exim4.conf.template.gmail` già configurato (si deve solo cambiare l'account di posta elettronica)

Inserisco anche una versione in inglese ([separate sources](#))

Step 1

Run `dpkg-reconfigure exim4-config`

- Choose mail sent by smarthost; received via SMTP or fetchmail
- Type System Mail Name: e.g. `company.com`
- Type IP Adresses to listen on for incoming SMTP connections: `127.0.0.1`
- Leave Other destinations for which mail is accepted blank
- Leave Machines to relay mail for: blank
- Type Machine handling outgoing mail for this host (smarthost): `smtp.gmail.com:587`
- Choose NO, don't hide local mail name in outgoing mail.

- Chose NO, don't keep number of DNS-queries minimal (Dial-on-Demand).
- Choose mbox
- Choose NO, split configuration into small files
- Mail for postmaster. Leaving blank will not cause any problems though it is not recommended

Step 2

- Open the file `/etc/exim4/exim4.conf.template`
- Find the line `.ifdef DCconfig_smarthost DCconfig_satellite` and add the following in that section

```
send_via_gmail:
    driver = manualroute
    domains = ! +local_domains
    transport = gmail_smtp
    route_list = * smtp.gmail.com
```

If you have any other smarthost defined with “domains = ! +local_domains” remove that smarthost.

- Find the “begin authenticators”. In that section add the following

```
gmail_login:
    driver = plaintext
    public_name = LOGIN
    client_send = : yourname@gmail.com : YourGmailPassword
```

Make sure you have no other authenticators with the same public_name (LOGIN). Comment them out if needed (Thanks Jakub for reminding me)

- Find the comment “`transport/30_exim4-config_remote_smtp_smarthost`”. In that section add

```
gmail_smtp:
    driver = smtp
    port = 587
    hosts_require_auth = $host_address
    hosts_require_tls = $host_address
```

Step 3

- Run `update-exim4.conf`
- Do `/etc/init.d/exim4 restart`

That should be it. You can test by using the command line mail client.

- Run `mail user@example.com`
- Give a subject and press enter
- Type something and press enter
- Type a single . (dot) and press enter
- Press enter for a blank CC:

Altra versione in inglese:

- First, enable POP for your gmail account. You do that in the “Forwarding and POP” section

of the settings. Strangely enough.

- Next, add a domain list to your exim configuration:

```
domainlist use_gmail_domains =  
/usr/local/etc/exim/gmail.domains
```

This domain list will hold the list of domains to send via gmail, one domain per line. If you don't have any to add now, create the file empty with touch(1) so that you don't forget later. Exim won't complain either way.

- Create an authenticator. Note that although we're using the plaintext mechanism here, we'll force TLS in the transport so your details will not get transferred in the clear:

```
gmail_login:  
  driver = plaintext  
  public_name = LOGIN  
  client_send = : YourGmailUsername@gmail.com : YourGmailPassword
```

Note that in a default exim configuration there are usually no authenticators, so don't forget the **begin authenticators** statement if this is your first one.

- Add a router:

```
send_via_gmail:  
  driver = manualroute  
  domains = +use_gmail_domains  
  transport = gmail_smtp  
  route_list = "*" smtp.gmail.com byname"
```

- Add a transport, forcing it to use AUTH and TLS:

```
gmail_smtp:  
  driver = smtp  
  hosts = smtp.gmail.com  
  hosts_require_auth = smtp.gmail.com  
  hosts_require_tls = smtp.gmail.com
```

That's all it requires. You may now need to lock down the permissions on your configuration file to stop anyone reading your username and password from it. Advanced exim users can work out how to put this information in a separate file easily enough.