Exim4
passo dopo passo

Configurazione graduale di Exim

a cura di

Stefano Sasso

stefano(at)gnustile.net

Versione 1.1 - 20 Agosto 2009

Indice

Prefazione

In questo libro vedremo la configurazione di Exim 4 (non la versione 3.x!) passo dopo passo, introducendo ogni volta nuovi elementi.

Questo libro non é un'introduzione a Exim, né vuole esserlo. Sono quindi necessarie alcune conoscenze di base, in quando non verrá spiegato né il funzionamento né la struttura di Exim.

Capitolo 1

Introduzione a Exim

Exim é un MTA, un Mail Transfer Agent, ovvero un server di posta elettronica. Esso si occupa di ricevere la posta, se é locale la conserva, altrimenti la invia al giusto server destinatario.

1.1 Il file di configurazione

Il file di configurazione si divide in varie parti, e puó essere suddiviso in piú files. Dal file principale é possibile includere altri files usando la direttiva *.include*

```
.include /etc/exim/spf-acl.conf
```

Le parti in cui si divide il file di configurazione sono

- Generale, la configurazione generale del servizio di posta
- *acl*, vengono definite le access control list
- *routers*, definisce che percorso deve intraprendere la mail in ingresso, a seconda di varie condizioni. É importante l'ordine di definizione. Il primo match vince.
- *transports*, fa effettivamente prendere una strada alle mail selezionate dai routers
- *retry*, regole per riprovare a inviare i messaggi in coda
- *rewrite*, regole per la riscrittura degli indirizzi
- *authenticators*, definisce le impostazioni per l'autenticazione smtp

Parte I

Consegna base della posta

Capitolo 2

Consegna locale

2.1 Configurazione di base

```
##########################################
# Configurazione generale                #
##########################################
primary_hostname = simplemx.dominio.net

# numero 0-16 che identifica il sistema
# in un cluster di posta (usato per generare ID di messaggi)
localhost_number = 12

qualify_domain = dominio.net
smtp_banner = $smtp_active_hostname ESMTP Exim\n$tod_full
never_users = root
hostlist   relay_from_hosts = 127.0.0.1 : 72.20.214.0/24
domainlist local_domains = @ : dominio.net : posta.dominio.net
host_lookup = *
rfc1413_hosts = *
rfc1413_query_timeout = 0s
message_size_limit = 50M
return_size_limit = 100K
smtp_accept_queue = 270
smtp_accept_max = 400
smtp_accept_max_per_host = 10
smtp_accept_reserve = 100
smtp_reserve_hosts = 127.0.0.1 : ::::1 : 72.20.214.0/24
queue_run_max = 16
ignore_bounce_errors_after = 3d
timeout_frozen_after = 3d
```

```
#definisce le acl da usare nelle varie situazioni

acl_smtp_helo = acl_check_helo
acl_smtp_rcpt = acl_check_rcpt
acl_smtp_data = acl_check_content

begin acl

##########################################
# Controllo sulla validità dell'HELO     #
##########################################
acl_check_helo:
  # accetta se arriva da pipe locale (no tcp/ip)
  accept hosts     = :
  # accetta se arriva da un host da cui è permesso il relay
  accept hosts     = +relay_from_hosts
  # droppa se ricevo come HELO il mio ip
  drop   condition = ${if match{$sender_helo_name}{MY_IP}{yes}{no} }
         message   = "Dropped spammer pretending to be us"
  # droppa se ricevo un ip come HELO
  drop   condition = ${if match{$sender_helo_name}{^[0-9]\.[0-9]\.[0-9]\.[0-9]}{yes}{no}}
         message   = "Dropped IP-only or IP-starting helo"
  accept

##########################################
# Controllo sulla validità dell'RCPT     #
##########################################
acl_check_rcpt:
  # accetta se arriva da pipe locale (no tcp/ip)
  accept  hosts         = :
  # nega il relay se l'indirizzo comincia con un .
  deny    local_parts   = ^.*[@%!/|] : ^\\.
  # accetta tutte le mail per postmaster locali
  accept  local_parts   = postmaster
          domains       = +local_domains
  # accetta le mail per i domini locali, dopo aver verificato il
  # recipient
  accept  domains       = +local_domains
          endpass
          verify        = recipient
  # accetta se il relay è consentito
  accept  hosts         = +relay_from_hosts
  # non consente il resto
  deny    message       = relay not permitted
```

```
##############################################
# Controllo sulla validità dei dati       #
##############################################
acl_check_content:
  accept

begin routers

##############################################
# Domini non locali                       #
# Invia la mail al giusto MX              #
##############################################
external_gw:
  driver    = dnslookup
  transport = remote_smtp
  domains   = ! +local_domains
  no_more

##############################################
# Alias di sistema                        #
# Cerca un alias nel file /etc/aliases    #
##############################################
system_aliases:
  driver         = redirect
  allow_fail
  allow_defer
  data           = ${lookup{$local_part}lsearch{/etc/aliases}}
  user           = mail
  group          = mail
  file_transport = address_file
  pipe_transport = address_pipe

##############################################
# Forward utente                          #
# "Esegue" il file .forward nella home    #
##############################################
userforward:
  driver          = redirect
  check_local_user
  file            = $home/.forward
  no_verify
  no_expn
  check_ancestor
# allow_filter
  file_transport  = address_file
```

```
  pipe_transport  = address_pipe
  reply_transport = address_reply
  condition       = ${if exists{$home/.forward} {yes} {no} }
  group           = mail

##########################################
# Utente di sistema                      #
# Invia la mail nella Maildir dell'utente #
##########################################
localuser:
  driver               = accept
  check_local_user
  transport            = local_delivery
  cannot_route_message = Unknown user


begin transports

##########################################
# PIPE Transport                         #
# Usato per chiamare programmi esterni   #
##########################################
address_pipe:
  driver = pipe
  return_output

##########################################
# FILE Transport                         #
# Usato per salvare su directory o file  #
##########################################
address_file:
  driver = appendfile
  delivery_date_add
  envelope_to_add
  return_path_add

##########################################
# REPLY Transport                        #
# Usato per autoreply                    #
##########################################
address_reply:
  driver = autoreply

##########################################
# SMTP Transport                         #
```

```
# Invia tramite SMTP                      #
##########################################
remote_smtp:
  driver = smtp

##########################################
# Local Delivery                         #
# Salva nella Maildir presente nella home #
##########################################
local_delivery:
  driver             = appendfile
  directory_mode     = 700
  group              = mail
  mode               = 0660
  maildir_format     = true
  directory          = ${home}/Maildir/
  create_directory   = true
  check_string       = ""
  escape_string      = ""
  mode_fail_narrower = false
  envelope_to_add    = true


begin retry

# This single retry rule applies to all domains and all errors. It specifies
# retries every 15 minutes for 2 hours, then increasing retry intervals,
# starting at 1 hour and increasing each time by a factor of 1.5, up to 16
# hours, then retries every 6 hours until 4 days have passed since the first
# failed delivery.

# Address or Domain    Error       Retries
# -----------------    -----       -------

*                      *           F,2h,5m; G,16h,1h,1.5; F,4d,6h
```

Da questo punto in poi non riscriveremo ogni volta il file di configurazione, ma scriveremo solo le parti diverse dalla configurazione qui vista.

2.2 Integrazione con procmail

Fondamentalmente esistono due modi per integrare Exim con Procmail: il primo consiste nell'inserire nel file *dot forward* di ogni utente una pipe verso procmail stesso, il secondo nel creare una pipe direttamente da Exim. Ecco come fare:

router:

```
procmail:
  debug_print = "R: procmail for $local_part@$domain"
  driver = accept
  domains = +local_domains
  check_local_user
  transport = procmail_pipe
  # emulate OR with "if exists"-expansion
  require_files = ${local_part}:\
                  ${if exists{/etc/procmailrc}\
                    {/etc/procmailrc}{${home}/.procmailrc}}:\
                  +/usr/bin/procmail
  no_verify
  no_expn
```

La parte *if exists...* indica di prendere in considerazione (per il require) il file */etc/procmailrc* se esiste, altrimenti */.procmailrc*.

transport:

```
procmail_pipe:
  debug_print = "T: procmail_pipe for $local_part@$domain"
  driver = pipe
  path = "/bin:/usr/bin:/usr/local/bin"
  command = "/usr/bin/procmail"
  return_path_add
  delivery_date_add
  envelope_to_add
```

Capitolo 3

Mascheramento verso internet

Ipotizziamo di avere un server che gestisce il dominio locale **rete.lan**, vogliamo che in uscita venga usato un altro mailserver, e che gli indirizzi vengano mascherati:
Inseriamo tra i router

```
smarthost:
  debug_print              = "R: smarthost for $local_part@$domain"
  driver                   = manualroute
  domains                  = ! +local_domains
  transport                = remote_smtp
  route_list               = * mailout-rr.mail.dominio.net
  host_find_failed         = defer
  same_domain_copy_routing = yes
  no_more
```

e poi

```
begin rewrite
  *@super.lan $1@external.net T
  *@+local_domains "${lookup{${local_part}}lsearch{/etc/exim4/email-rewrite}\
    {$value}fail}" Ffrs
```

e nel file */etc/exim4/email-rewrite*

```
user: myuser@isp1.com
other: otheruser@otherisp.net
```

Tuttavia se il nostro *smarthost* richiede autenticazione SMTP è necessaria una piccola modifica alla configurazione:

Aggiungiamo le seguenti macro all'inizio della configurazione:

```
AUTH_CLIENT_DATA = /etc/exim4/client_smtp_auth.txt
```

```
AUTH_CLIENT_USERNAME = ${extract{user}{AUTH_CLIENT_SEND_DATA}}
AUTH_CLIENT_PASSWORD = ${extract{pass}{AUTH_CLIENT_SEND_DATA}}
AUTH_CLIENT_REQUIRED = ${filter{${readfile{AUTH_CLIENT_DATA}{:}}}\
  {match{$item}{\N^\s*\d{1,3}(?:\.\d{1,3}){3}(?:/[0-9]{1,2})?\s*$\N}}}
AUTH_CLIENT_REQUIRE_SSL = ${filter{${sg{${filter{<\n${readfile{AUTH_CLIENT_DATA}}}\
  {match{${extract{require_ssl}{$item}}}{\N^(?i)\s*(true|yes|1)\s*$\N}}}}\
  {\N\n\N}{:}}}{match{$item}{\N^\s*\d{1,3}(?:\.\d{1,3}){3}\s*$\N}}}
AUTH_CLIENT_SEND_DATA = ${lookup{$host_address}iplsearch{AUTH_CLIENT_DATA}}
AUTH_CLIENT_ENABLED_PLAIN = ${if match{${extract{type}{AUTH_CLIENT_SEND_DATA}}}
  {\N^(?i)(.+,)*plain(,.+)*$\N}{true}{false}}
AUTH_CLIENT_ENABLED_LOGIN = ${if match{${extract{type}{AUTH_CLIENT_SEND_DATA}}}\
  {\N^(?i)(.+,)*login(,.+)*$\N}{true}{false}}
AUTH_CLIENT_ENABLED_CRAM = ${if match{${extract{type}{AUTH_CLIENT_SEND_DATA}}}\
  {\N^(?i)(.+,)*cram(,.+)*$\N}{true}{false}}
AUTH_CLIENT_SEND_CRAM_USER = AUTH_CLIENT_USERNAME
AUTH_CLIENT_SEND_CRAM_PASS = AUTH_CLIENT_PASSWORD
AUTH_CLIENT_SEND_LOGIN = : AUTH_CLIENT_USERNAME : AUTH_CLIENT_PASSWORD
AUTH_CLIENT_SEND_PLAIN = ^AUTH_CLIENT_USERNAME^AUTH_CLIENT_PASSWORD
```

Aggiungiamo al *transport* **remote_smtp**:

```
hosts_require_tls = AUTH_CLIENT_REQUIRE_SSL
hosts_require_auth = AUTH_CLIENT_REQUIRED
```

E nella sezione *authenticators*:

```
CRAM:
driver = cram_md5
public_name = CRAM-MD5
client_condition = AUTH_CLIENT_ENABLED_CRAM
client_name = AUTH_CLIENT_SEND_CRAM_USER
client_secret = AUTH_CLIENT_SEND_CRAM_PASS

LOGIN:
driver = plaintext
client_condition = AUTH_CLIENT_ENABLED_LOGIN
client_send = AUTH_CLIENT_SEND_LOGIN

PLAIN:
driver = plaintext
client_condition = AUTH_CLIENT_ENABLED_PLAIN
client_send = AUTH_CLIENT_SEND_PLAIN
```

Il file */etc/exim4/client_smtp_auth.txt* conterrà un elenco

```
192.168.0.0/16: require_ssl="true" type="cram" user="myuser" pass="mypass"
99.11.12.0/24: require_ssl="true" type="login,plain" user="myuser123" pass="mypass123"
```

È anche possibile usare il nome dello smarthost invece dei suoi indirizzi ip, basta fare una ricerca wildcard di *$host* invece di una ricerca ip per *$host_address*:

```
AUTH_CLIENT_SEND_DATA = ${lookup{$host}nwildlsearch{AUTH_CLIENT_DATA}}
```

Se siamo nella situazione in cui utilizziamo un servizio di antivirus/antispam esterno, e vogliamo far passare per esso anche le mail intra-dominio, possiamo pensare di utilizzare un apposito header (o una macro acl) sui messaggi (locali) e quindi, se tale header non è presente, mandare allo *smarthost* anche i messaggi interni.

Parte II

Altre ACL

Capitolo 4

ACL anti-spam

Aggiungiamo ora, a quelle giá viste, qualche altra ACL che puó esserci utile per ridurre lo spam in ingresso.

4.1 ACL generali

```
############################################
# Controllo sulla validità dell'HELO      #
############################################
acl_check_helo:
  # accetta se arriva da pipe locale (no tcp/ip)
  accept hosts     = :
  # accetta se arriva da un host da cui è permesso il relay
  accept hosts     = +relay_from_hosts
  # droppa se ricevo come HELO il mio ip
  drop   condition = ${if match{$sender_helo_name}{MY_IP}{yes}{no}}
         message   = "Dropped spammer pretending to be us"
  # droppa se ricevo un ip come HELO
  drop   condition = ${if match{$sender_helo_name}{^[0-9]\.[0-9]\.[0-9]\.[0-9]}{yes}{no}}
         message   = "Dropped IP-only or IP-starting helo"

  # NUOVE CONDIZIONI:
  # helo non valido (RFC2821 4.1.3)
  drop   condition = ${if isip{$sender_helo_name}}
         message   = Access denied - Invalid HELO name (See RFC2821 4.1.3)
  # helo non fqdn
  drop   condition = ${if match{$sender_helo_name}{\N^\[\N}{no}{yes}}
         condition = ${if match{$sender_helo_name}{\N\.\N}{no}{yes}}
         message   = Access denied - Invalid HELO name (See RFC2821 4.1.1.1)
  drop   condition = ${if match{$sender_helo_name}{\N\.$\N}}
         message   = Access denied - Invalid HELO name (See RFC2821 4.1.1.1)
```

```
  drop   condition = ${if match{$sender_helo_name}{\N\.\.\N}}
         message   = Access denied - Invalid HELO name (See RFC2821 4.1.1.1)
  # helo è il mio hostname
  drop   message   = "REJECTED - Bad HELO - Host impersonating [$sender_helo_name]"
         condition = ${if match{$sender_helo_name}{$primary_hostname}{yes}{no}}
  # helo è uno dei domini gestiti da me
  drop   message   = "REJECTED - Bad HELO - Host impersonating [$sender_helo_name]"
         condition = ${if match_domain{$sender_helo_name}{+local_domains}{true}{false}}
  # rate limit, al massimo 1000 email per ora da un host
  defer  message   = Sender rate exceeds $sender_rate_limit messages \
                       per $sender_rate_period
         ratelimit = 1000 / 1h / per_conn / leaky / $sender_host_address

  accept

##############################################
# Controllo sulla validità dell'RCPT      #
##############################################
acl_check_rcpt:
  # accetta se arriva da pipe locale (no tcp/ip)
  accept  hosts         = :
  # nega il relay se l'indirizzo comincia con un .
  deny    local_parts   = ^.*[@%!/|] : ^\\.

  # NUOVE CONDIZIONI:
  # i messaggi bounce da postmaster@ sono inviate solo ad un indirizzo
  drop    message       = Legitimate bounces are never sent to more than one recipient.
          senders       = : postmaster@*
          condition     = ${if >{$recipients_count}{1}{true}{false}}
  # cancella se ci sono più di 5 destinazioni fallite
  drop    message       = REJECTED - Too many failed recipients - count = $rcpt_fail_count
          log_message   = REJECTED - Too many failed recipients - count = $rcpt_fail_count
          condition     = ${if > {${eval:$rcpt_fail_count}}{5}{yes}{no}}
          !verify       = recipient/callout=2m,defer_ok,use_sender
  # cancella se una delle destinazioni è una spamtrap
  drop    condition     = ${lookup{$local_part@$domain}lsearch{/etc/exim/spamtraps} {yes}{no}}
          logwrite      = :main,reject: $sender_host_address - $local_part@$domain is a trap
          message       = I don't think so
  # cancella se la destinazione è protetto.com e la sorgente è diversa da miodominio.com
  deny    log_message   = $sender_address is not permitted to send to \
                            protetto.com o protetto2.com
          domains       = protetto.com : protetto2.com
          ! senders     = *miodominio.com

  # VECCHIE CONDIZIONI:
```

```
  # accetta tutte le mail per postmaster locali
  accept  local_parts   = postmaster
          domains       = +local_domains
  # accetta le mail per i domini locali, dopo aver verificato il
  # recipient
  accept  domains       = +local_domains
          endpass
          verify        = recipient
  # accetta se il relay è consentito
  accept  hosts         = +relay_from_hosts
  # non consente il resto
  deny    message       = relay not permitted

##############################################
# Controllo sulla validità dei dati       #
##############################################
acl_check_content:
  # blocca se sia il soggetto che il testo sono vuoti
  deny    message       = REJECTED - No Subject nor body
          !condition    = ${if def:h_Subject:}
          condition     = ${if <{$body_linecount}{1}{true}{false}}
  # blocca i messaggi con proplemi mime
  deny    message       = This message contains a MIME error ($demime_reason)
          demime        = *
          condition     = ${if >{$demime_errorlevel}{2}{1}{0}}
  # blocca i messaggi con determinate estensioni
  deny    message       = This message contains an unwanted \
                            file extension ($found_extension)
          demime        = scr:vbs:bat:lnk:pif:vbe:reg
  # accetta il resto
  accept
```

4.2 DNS-RBL

Inserire all'inizio dell'acl di RCPT:

```
drop    message  = REJECTED - ${sender_host_address} is blacklisted at \
                     $dnslist_domain ($dnslist_value); ${dnslist_text}
        dnslists = sbl-xbl.spamhaus.org/<;$sender_host_address;$sender_address_domain

drop    message  = REJECTED - ${sender_address_domain} is blacklisted at \
                     ${dnslist_domain}; ${dnslist_text}
        dnslists = nomail.rhsbl.sorbs.net/$sender_address_domain
```

```
drop    message  = REJECTED - ${sender_host_address} is blacklisted at \
                     ${dnslist_domain}; ${dnslist_text}
        dnslists = zen.spamhaus.org : bl.spamcop.net : cbl.abuseat.org : list.dsbl.org
```

4.3 Homemade Greylisting

4.3.1 Database

```
CREATE TABLE exim_greylist
(
  id int(11) NOT NULL auto_increment PRIMARY KEY,
  relay_ip varchar(21),
  from_domain varchar(85),
  block_expires datetime NOT NULL,
  record_expires datetime NOT NULL,
  origin_type enum('MANUAL','AUTO') NOT NULL DEFAULT 'AUTO',
  create_time datetime NOT NULL,
  KEY exim_lookup (relay_ip,from_domain)
);
```

4.3.2 Configurazione Exim

```
GREYLIST_TEST = SELECT IF(NOW() > block_expires, 2, 1) \
  FROM exim_greylist \
  WHERE relay_ip = '${quote_mysql:$sender_host_address}' \
  AND from_domain = '${quote_mysql:$sender_address_domain}' \
  AND record_expires > NOW()

GREYLIST_ADD = \
  INSERT INTO exim_greylist \
  SET relay_ip   = '${quote_mysql:$sender_host_address}', \
  from_domain    = '${quote_mysql:$sender_address_domain}', \
  block_expires  = DATE_ADD(NOW(), INTERVAL 10 MINUTE), \
  record_expires = DATE_ADD(NOW(), INTERVAL 28 DAY), \
  origin_type    = 'AUTO', \
  create_time    = NOW()

GREYLIST_UPDATE = \
  UPDATE exim_greylist \
  SET    record_expires = DATE_ADD(now(), INTERVAL 28 DAY) \
  WHERE relay_ip        = '${quote_mysql:$sender_host_address}' \
  AND    from_domain    = '${quote_mysql:$sender_address_domain}' \
  AND    record_expires > NOW()
```

nelle ACL RCPT:

```
warn
  set acl_m2      = ${lookup mysql{GREYLIST_TEST}{$value}{0}}
defer
  ! hosts         = +whitelist
  ! hosts         = +relay_from_hosts
  ! authenticated = *
  condition       = ${if eq{$acl_m2}{0}{yes}}
  condition       = ${lookup mysql{GREYLIST_ADD}{yes}{no}}
  message         = Now greylisted - please try again in five minutes.
defer
  ! hosts         = +whitelist
  ! hosts         = +relay_from_hosts
  ! authenticated = *
  condition       = ${if eq{$acl_m2}{1}{yes}}
  message         = Still greylisted - please try again in five minutes.
defer
  ! hosts         = +whitelist
  ! hosts         = +relay_from_hosts
  ! authenticated = *
  condition       = ${lookup mysql{GREYLIST_UPDATE}{no}{no}}
  message         = Greylist update failed
```

4.4 Homemade Blacklisting

```
BLACKLIST_TEST = SELECT 1 \
        FROM   exim_blacklist \
        WHERE  relay_ip = '${quote_mysql:$sender_host_address}' \
        AND    NOW() < expires

BLACKLIST_ADD = INSERT INTO exim_blacklist \
        SET relay_ip  = '${quote_mysql:$sender_host_address}', \
            expires   = DATE_ADD(NOW(), INTERVAL 1 DAY), \
            created   = NOW(), \
            sender    = '${quote_mysql:$sender_address}', \
            recipient = '${quote_mysql:$original_local_part@$original_domain}'

BLACKLIST_UPDATE = UPDATE exim_blacklist \
        SET   expires = DATE_ADD(NOW(), INTERVAL 1 WEEK) \
        WHERE relay_ip = '${quote_mysql:$sender_host_address}'
```

e nelle ACL:

```
warn
```

```
  set acl_m3      = ${lookup mysql{BLACKLIST_TEST}{$value}{0}}
deny
  ! hosts         = +whitelist_hosts
  ! senders       = +whitelist_users
  ! authenticated = *
  condition       = ${if eq{$acl_m3}{1}{yes}}
  condition       = ${lookup mysql{BLACKLIST_UPDATE}{yes}{yes}}
  message         = You are still blacklisted for hitting a spam trap
deny
  ! hosts         = +whitelist_hosts
  ! senders       = +whitelist_users
  ! authenticated = *
# recipients      = trap@dominio.com
  condition       = ${lookup{$local_part@$domain}lsearch{/etc/exim/spamtraps} {yes}{no}}
  condition       = ${lookup mysql{BLACKLIST_ADD}{yes}{yes}}
  message         = You are now blacklisted for hitting a spam trap (1)
```

Parte III

Scansione Antivirus/Antispam

Capitolo 5

exiscan-acl

exiscan-acl é una serie di estensioni di exim che consentono la scansione antispam/antivirus direttamente nelle direttive acl.

5.1 Configurazione di exim con exiscan-acl

Vediamo subito un esempio:

```
av_scanner    = clamd:192.168.177.44 3310
spamd_address = 192.168.177.45 783
begin acl
 acl_check_data:
  # antivirus
  deny message   = This message contains malware ($malware_name)
       demime    = *
       malware   = *
  deny message   = Message scored $spam_score spam points.
       condition = ${if <{$message_size}{150k}{1}{0}}
       spam      = nobody:true
       condition = ${if >{$spam_score_int}{150}{1}{0}}
```

È tuttavia possibile evitare il blocco a smtp-time per gestire lo spam e i virus successivamente. Basta utilizzare qualche piccolo trucchetto, come l'utilizzo di una macro acl o di una intestazione personalizzata; vediamo i due esempi (in questo caso solo per antivirus):

```
warn set acl_m5 = virus_found
     set acl_m6 = $malware_name
     demime = *
     malware = *
```

e

```
warn message = X-Virus-found: true
     message = X-Virus-name: $malware_name
     demime = *
     malware = *
```

Capitolo 6

AMaViS

AMaViS, *A Mail Virus Scanner* é un altro metodo che possiamo utilizzare per effettuare una scansione antivirus e antispam. A differenza di *exiscan-acl* peró, in questo caso exim tratterá ogni mail due volte, con evidente aumento del carico.

6.1 Configurazione di Exim

Exim deve rimanere in ascolto anche su 127.0.0.1:10025

```
# router
amavis:
  driver     = manualroute
  condition  = ${if eq {$interface_port}{10025} {0}{1}}
  # scansione solo delle mail in ingresso
  domains    = +local_domains
  transport  = amavis
  route list = * amavis-1.mail.dominio.net byname
  self       = send

# transport
amavis:
  driver = smtp
  port   = 10024
  allow_localhost
```

6.2 Alcune parti della configurazione di AMaViS

```
$mydomain = 'mail.dominio.net';
$forward_method = 'smtp:127.0.0.1:10025'; #overridden by relayhost_is_client
$relayhost_is_client = 1;
```

```
$notify_method = $forward_method;
$inet_socket_bind = 'AA.BB.CC.DD';
@inet_acl = qw( AA.BB.CC.DA AA.BB.CC.DB AA.BB.CC.DC );
```

Capitolo 7

SpamAssassin

7.1 Exim e SpamAssassin

Un ulteriore metodo per exim di invocare SpamAssassin é quello di chiamare direttamente il client *spamc*.

```
# router
spamcheck_router:
  no_verify
  check_local_user
  # When to scan a message :
  # -   it isn't already flagged as spam
  # -   it isn't already scanned
  condition = "${if and { {!def:h_X-Spam-Flag:} \
                {!eq {$received_protocol}{spam-scanned}}} \
                {1}{0}}
  driver    = accept
  transport = spamcheck

# transport
spamcheck:
  driver             = pipe
  command            = /usr/bin/exim4 -oMr spam-scanned -bS
  use_bsmtp          = true
  transport_filter   = /usr/bin/spamc
  home_directory     = "/tmp"
  current_directory  = "/tmp"
  # must use a privileged user to set $received_protocol on the way back in!
  user               = Debian-exim
  group              = Debian-exim
  log_output         = true
  return_fail_output = true
```

```
return_path_add    = false
message_prefix     =
message_suffix     =
```

Con questo modo è possibile utilizzare configurazioni avanzate di spamassassin, ad esempio configurazioni personalizzate a seconda del destinatario della mail (è possibile salvare tali informazioni anche su db mysql o su ldap).
In questo caso il *transport_filter* deve essere

```
transport_filter   = /usr/bin/spamc -u '${local_part}@${domain}'
```

Riferirsi al sito di spamassassin per la configurazione dello stesso.[1]

7.2 SA-Exim

Da qualche versione a questa parte é possibile invocare SpamAssassin in SMTP-time utilizzando le estensioni *local scan* con **sa-exim**[2].
É quindi sufficiente inserire in *exim4.conf*

```
local_scan_path = /usr/lib/exim4/local_scan/sa-exim.so
```

e configurare **sa-exim** da *sa-exim.conf*.
Se invece vogliamo evitare la scansione per alcune destinazioni (es: postmaster) è sufficiente inserire nelle ACL rcpt:

```
warn   message       = X-SA-Do-Not-Rej: Yes
       local_parts   = +nosarej:postmaster:abuse

warn   message       = X-SA-Do-Not-Run: Yes
       hosts         = +relay_from_hosts

warn   message       = X-SA-Do-Not-Run: Yes
       authenticated = *
```

o, in alternativa, in *sa-exim.conf*

```
SAEximRunCond: ${if !eq {$acl_m0}{do-not-scan} {1} {0}}
SAEximRejCond: ${if !eq {$acl_m0}{do-not-reject} {1} {0}}
```

e nelle ACL rcpt:

```
##### Checks for postmaster or abuse - we'll scan, still, but not reject
##### Don't reject for certain users
warn   local_parts   = postmaster : abuse
```

[1] http://wiki.apache.org/spamassassin/UsingSQL
[2] http://marc.merlins.org/linux/exim/sa.html

```
        set acl_m0    = do-not-reject

  ##### Check for situations we don't even scan (local mail)
  ##### Don't scan if hosts we relay for (probably dumb MUAs),
  warn   hosts         = +relay_from_hosts:127.0.0.1/8
         set acl_m0    = do-not-scan

  ##### Don't scan non-smtp connections (empty host list)
  warn   hosts         = :
         set acl_m0    = do-not-scan

  ##### Don't scan if authenticated
  warn   authenticated = *
         set acl_m0    = do-not-scan
```

7.3 Exim e SAlearn

É possibile creare una serie di indirizzi e-mail a cui inviare spam/ham, in modo da istruire al meglio spamassassin:

```
#router
learn_spam_ham_router:
  driver      = accept
  transport   = learn_spam_ham
  local_parts = spam : ham
  #condition  = ${lookup {$domain} lsearch {/etc/exim/salearn_domains}{yes}{no}}
  domains     = spam.dominio.net

#transport
learn_spam_ham:
  driver  = pipe
  command = /usr/bin/sa-learn --${local_part} --no-rebuild --single
  user    = mail
  group   = mail
```

Parte IV

Autenticazione SMTP

Capitolo 8

Autenticazione SMTP

Innanzitutto dobbiamo aggiungere una riga simile alla seguente alle ACL RCPT:

```
accept authenticated = *
```

8.1 Autenticazione SMTP con Courier

```
begin authenticators
plain:
  driver           = plaintext
  public_name      = PLAIN
  server_prompts   = :
  server_condition = ${if eq{${readsocket{COURIERSOCKET}{AUTH \
                       ${eval:13+${strlen:$2$3}}\nexim\nlogin\n$2\n$3\n} \
                       {5s}{ } }}{FAIL }{no}{yes}}
  # impostare server_set_id solo se vogliamo sovrascrivere il
  # mittente con l'utente dell'auth smtp
  # server_set_id    = $2

login:
  driver           = plaintext
  public_name      = LOGIN
  server_prompts   = Username:: : Password::
  server_condition = ${if eq{${readsocket{COURIERSOCKET}{AUTH \
                       ${eval:13+${strlen:$1$2}}\nexim\nlogin\n$1\n$2\n} \
                       {5s}{ } }}{FAIL }{no}{yes}}
  # impostare server_set_id solo se vogliamo sovrascrivere il
  # mittente con l'utente dell'auth smtp
  # server_set_id    = $1
```

8.2 Autenticazione SMTP con MySQL

Ovviamente quanto vedremo potrá tranquillamente essere adattato anche a PostgreSQL e simili.

```
begin authenticators
plain:
  driver           = plaintext
  public_name      = PLAIN
  server_prompts   = :
  server_condition = "${if and { \
                        {!eq{$2}{}} \
                        {!eq{$3}{}} \
                        {crypteq{$3}{${lookup mysql{SELECT password FROM users WHERE ( \
                        domain = ${quote_mysql:${domain:$2}} \
                        AND user = ${quote_mysql:${local_part:$2}}) \
                        OR email = ${quote_mysql:$2} }{$value}fail}} }} \
                        {yes}{no}}"
  # impostare server_set_id solo se vogliamo sovrascrivere il
  # mittente con l'utente dell'auth smtp
  # server_set_id    = $2

login:
  driver           = plaintext
  public_name      = LOGIN
  server_prompts   = "Username:: : Password::"
  server_condition = "${if and { \
                        {!eq{$1}{}} \
                        {!eq{$2}{}} \
                        {crypteq{$2}{${lookup mysql{SELECT password FROM users WHERE ( \
                        domain = ${quote_mysql:${domain:$1}} \
                        AND user = ${quote_mysql:${local_part:$1}}) \
                        OR email = ${quote_mysql:$1} }{$value}fail}} }} \
                        {yes}{no}}"
  # impostare server_set_id solo se vogliamo sovrascrivere il
  # mittente con l'utente dell'auth smtp
  # server_set_id    = $1
```

8.3 Autenticazione SMTP con un server IMAP

Vedremo ora come collegare l'autenticazione SMTP con l'autenticazione IMAP. Questa parte é interessante perché vedremo anche l'integrazione di exim con perl.
Creiamo subito il file */etc/exim/exim_perl.pl*

```
#!/usr/bin/perl
```

```
use Net::IMAP::Simple;

sub imaplogin
{
  my $host = shift;
  my $account = shift;
  my $password = shift;

  # open a connection to the imap server
  if (! ($server = new Net::IMAP::Simple($host)))
  {
    return 0;
  }

  # login, if success return 1 (true) else 0 (false)
  if ($server->login( $account, $password ))
  {
    return 1;
  }
  else
  {
    return 0;
  }

  server->close();
}
```

E all'inizio della configurazione di exim inseriamo:

```
perl_startup = do '/etc/exim/exim_perl.pl'
perl_at_start
```

Mentre nella parte relativa all'autenticazione smtp inseriamo

```
begin  authenticators

plain:
  driver           = plaintext
  public_name      = PLAIN
  server_condition = ${perl{imaplogin}{localhost}{$2}{$3}}
  # impostare server_set_id solo se vogliamo sovrascrivere il
  # mittente con l'utente dell'auth smtp
  # server_set_id    = $2

login:
```

```
driver           = plaintext
public_name      = LOGIN
server_prompts   = "Username:: : Password::"
server_condition = ${perl{imaplogin}{localhost}{$1}{$2}}
# impostare server_set_id solo se vogliamo sovrascrivere il
# mittente con l'utente dell'auth smtp
# server_set_id    = $1
```

Parte V

Connessioni sicure

Capitolo 9

SSL e TLS

dopo aver generato i certificati[1], modifichiamo la parte ″globale″ della configurazione di exim:

```
tls_advertise_hosts  = *
tls_certificate      = /etc/ssl/exim.crt
tls_privatekey       = /etc/ssl/exim.pem
daemon_smtp_ports    = 25 : 465 : 587
tls_on_connect_ports = 465
auth_advertise_hosts = *
```

[1] */usr/share/doc/exim4/examples/exim-gencert* ci puó aiutare

Parte VI

Utenti/Domini Virtuali

Capitolo 10

Domini virtuali da file di testo

10.1 Domini virtuali su utenti locali

Vediamo ora dei domini (e alias) virtuali, che inviano ad utenti locali:
creiamo il file **/etc/exim4/vdomains** con al suo interno:

```
vdomain1.net
vdomain2.net
```

quindi modifichiamo la configurazione di exim:

```
domainlist local_domains = dominio.net : posta.dominio.net : \
                           lsearch;/etc/exim4/vdomains
```

Aggiungiamo ora in prima posizione questo router:

```
virtual_domain_aliases:
  driver = redirect
  domains = lsearch;/etc/exim4/vdomains
  data = ${lookup{$local_part}lsearch{/etc/exim4/aliases-$domain}}
  cannot_route_message = Unknown vdomain-text user
  headers_add = X-virtual-user: $local_part\n\
                X-virtual-domain: $domain\n\
                X-virtual-address: $local_part@$domain\n\
                X-mailhub-route: $primary_hostname
  no_more
```

10.2 Utenti virtuali con Dovecot

Configuriamo subito Dovecot per l'autenticazione su file di testo:

```
auth default {
  userdb static {
    args = uid=500 gid=500 home=/home/dovecot/%d/%n
  }
  passdb passwd-file {
    args = /home/dovecot/passwd
  }
}
```

proseguiamo con la configurazione di exim: il router

```
dovecot_router:
  driver = accept
  require_files = +/home/dovecot/$domain/$local_part/
  transport = dovecot_transport
```

e il transport:

```
dovecot_transport:
  driver = appendfile
  user = dovecot
  group = dovecot
  mode = 0600
  directory = /home/dovecot/${lc:$domain}/${lc:$local_part}/Maildir/
  maildir_format = true
  mode_fail_narrower = false
  envelope_to_add = true
  return_path_add = true
```

Capitolo 11

Domini virtuali da server SQL

Vedremo qui una configurazione con server MySQL. Ovviamente é facilmente adattabile anche a PostgreSQL.

11.1 Configurazione Database

```
CREATE TABLE aliases (
  aliasid bigint(20) NOT NULL auto_increment,
  aliasname varchar(100) NOT NULL,
  domain varchar(100) NOT NULL,
  destination varchar(201) NOT NULL,
  active tinyint(1) NOT NULL default '1',
  PRIMARY KEY  (aliasid)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;

CREATE TABLE domains (
  id int(11) NOT NULL auto_increment,
  domain varchar(100) NOT NULL,
  created datetime NOT NULL,
  catchall varchar(201),
  active tinyint(1) NOT NULL default '1',
  PRIMARY KEY  (id),
  UNIQUE KEY domain (domain)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;

CREATE TABLE users (
  userid bigint(20) NOT NULL auto_increment,
  username varchar(100) NOT NULL,
  domain varchar(100) NOT NULL,
  email varchar(201) NOT NULL,
  password varchar(200) NOT NULL,
```

```
  maildir varchar(255) NOT NULL,
  active tinyint(1) NOT NULL default '1',
  quota varchar(50) NOT NULL,
  created datetime NOT NULL,
  vacation tinyint(1) NOT NULL default '0',
  vacationmsg text NOT NULL,
  vacationsubj varchar(250) NOT NULL,
  PRIMARY KEY  (userid)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

11.2 Configurazione Exim

Nel file di configurazione, prima di qualsiasi altra cosa inseriamo

```
hide mysql_servers = "127.0.0.1/exim/exim/exim"
#hide mysql_servers = "<host>/<database>/<user>/<password>"
#hide mysql_servers = "myslaves-rr.mailcluster.gs.net/mailc_db/user/pass" : \
#                     "mymaster-rr.mailcluster.gs.net/mailc_db/user/pass" : \
#                     "mymaster-1.mailcluster.gs.net/mailc_db/user/pass" : \
#                     "mymaster-2.mailcluster.gs.net/mailc_db/user/pass"

Q_LOCAL_DOMAIN=SELECT domain FROM domains WHERE domain=\'$domain\'
Q_MAIL_ALIAS=SELECT destination FROM aliases WHERE \
               aliasname=\'$local_part\' AND domain=\'$domain\'
Q_VALID_EMAIL=SELECT userid FROM users WHERE email=\'$local_part@$domain\'
Q_MAIL_BOX_DIR=SELECT maildir FROM users WHERE email=\'$local_part@$domain\'
Q_MAIL_BOX_QUOTA=SELECT quota FROM users WHERE email=\'$local_part@$domain\'
```

dopodiché modifichiamo gli elenchi dei domini in modo che risultino simili a questi:

```
domainlist local_domains = @ : dominio.net : posta.dominio.net : lsearch;/etc/exim4/vdomains
domainlist virtual_domains = mysql;Q_LOCAL_DOMAIN
domainlist my_domains = +local_domains : +virtual_domains
```

Avremo poi cura di sistemare le acl in modo che rispecchino i nuovi domini da accettare, e nei *router* per la posta locale (non virtuale) inseriremo il selettore

```
domains       = +local_domains
```

Inseriamo quindi dei router per i nostri utenti o alias virtuali

```
virtual_mysql_aliases:
  driver = redirect
  domains = +virtual_domains
  data = ${lookup mysql{Q_MAIL_ALIAS}{$value}}
```

```
  #rewrite = true
  user = mail
  local_part_suffix = +*
  local_part_suffix_optional
  file_transport = address_file
  pipe_transport = address_pipe
  headers_add = X-virtual-user: $local_part\n\
                X-virtual-domain: $domain\n\
                X-virtual-transport: mysql-alias\n\
                X-virtual-address: $local_part@$domain\n\
                X-mailhub-route: $primary_hostname

virtual_mysql_mailbox:
  driver = accept
  domains = +virtual_domains
  local_part_suffix = +*
  local_part_suffix_optional
  condition = ${lookup mysql{Q_VALID_EMAIL}}
  transport = virtual_mysql_delivery
  headers_add = X-virtual-user: $local_part\n\
                X-virtual-domain: $domain\n\
                X-virtual-transport: mysql-maildir\n\
                X-virtual-address: $local_part@$domain\n\
                X-mailhub-route: $primary_hostname
```

e ovviamente inseriamo anche il transport appropriato:

```
virtual_mysql_delivery:
  driver = appendfile
  directory = /srv/vmail/${lookup mysql{Q_MAIL_BOX_DIR}{$value}}
  maildir_format
  create_directory = true
  quota = ${lookup mysql{Q_MAIL_BOX_QUOTA}{$value}{5M}}
  user = mail
  group = mail
  mode = 0660
  directory_mode = 0770
```

Nel caso implementassimo una soluzione antispam potrebbe essere utile inviare tutto lo spam ricevuto in una cartella diversa dalla **inbox**, per questo prima del router *virtual_mysql_mailbox* inseriamo un altro router:

```
virtual_mysql_spam_mailbox:
  driver = accept
  domains = +virtual_domains
  local_part_suffix = +*
```

```
  local_part_suffix_optional
  condition = ${lookup mysql{Q_VALID_EMAIL}}
  condition = ${if eq {${if def:h_X-SpamFolder {true}{false}}} {true}}
  transport = virtual_mysql_spam_delivery
  headers_add = X-virtual-user: $local_part\n\
                X-virtual-domain: $domain\n\
                X-virtual-transport: mysql-maildir\n\
                X-virtual-address: $local_part@$domain\n\
                X-mailhub-route: $primary_hostname
```

e inseriamo poi un altro transport:

```
virtual_mysql_spam_delivery:
  driver = appendfile
  directory = /srv/vmail/${lookup mysql{Q_MAIL_BOX_DIR}{$value}}/.Junk
  maildir_format
  create_directory = true
  quota = ${lookup mysql{Q_MAIL_BOX_QUOTA}{$value}{5M}}
  user = mail
  group = mail
  mode = 0660
  directory_mode = 0770
```

11.2.1 Controllo della validitá di un utente/alias virtuale da ACL

É possibile effettuare un controllo sulla validitá di un utente o alias virtuale in fase di acl, e di mandare un messaggio personalizzato nel caso questo non esista.
Inseriamo innanzitutto, all'inizio della configurazione

```
Q_COUNT_MAIL=SELECT COUNT(*) FROM users WHERE email=\'$local_part@$domain\'
Q_COUNT_ALIAS=SELECT COUNT(*) FROM aliases WHERE aliasname=\'$local_part\' \
                AND domain=\'$domain\'
```

e poi, nelle acl, dobbiamo modificare

```
  accept  domains       = +my_domains
          endpass
          verify        = recipient
```

in

```
  accept  domains       = +local_domains
          endpass
          verify        = recipient
```

facendo in modo che controlli il recipiend solo dei domini non virtuali; e aggiungendo subito dopo

```
  deny message = Unknown virtual user/alias
       domains = +virtual_domains
       condition = ${if and {{eq {${lookup mysql{Q_COUNT_MAIL}}}{0}} \
                     {eq {${lookup mysql{Q_COUNT_ALIAS}}}{0}}}}

  accept domains = +virtual_domains
         endpass
```

11.3 Qualcosa in piú...

É possibile introdurre poi tutti i controlli antispam che abbiamo visto in precedenza: spamassassin, exiscan, dnsrbl, blacklist, greylist...
Vedremo peró ora cosa si puó aggiungere per rendere piú professionale i nostri domini virtuali: proveremo ad aggiungere un autorisponditore nel caso la casella sia ″in vacanza″ e la gestione delle caselle catchall per i vari domini.

11.3.1 vacation

Inseriamo all'inizio della configurazione

```
Q_ISAWAY=SELECT domain FROM users WHERE domain='${quote_mysql:$domain}' \
           AND username='${quote_mysql:$local_part}' AND vacation=1
Q_AWAYSUBJ=SELECT vacationsubj FROM users WHERE domain='${quote_mysql:$domain}' \
             AND username='${quote_mysql:$local_part}'
Q_AWAYTEXT=SELECT vacationmsg FROM users WHERE domain='${quote_mysql:$domain}' \
             AND username='${quote_mysql:$local_part}'
```

poi, come router, prima del router per il delivery normale, inseriamo

```
vacation_router:
  driver = accept
  domains = ${lookup mysql {Q_ISAWAY}{$value}}
  transport = vacation_autoreply
  unseen
```

e nei transport

```
vacation_autoreply:
  driver = autoreply
  to = ${sender_address}
  reply_to = "${local_part}@${domain}"
  from = "vacation@${domain}"
  subject = ${lookup mysql {Q_AWAYSUBJ}{$value} \
              {Automatic reply from ${local_part}@${domain}}}
  text = ${lookup mysql {Q_AWAYTEXT}{$value}}
```

11.3.2 Catch-All

Aggiungiamo all'inizio della configurazione

```
Q_CATCHALL=SELECT catchall FROM domains WHERE domain='${quote_mysql:$domain}'
```

poi, dopo i vari router per i domini virtuali

```
mysql_catchall:
  driver = redirect
  domains = +virtual_domains
  file_transport = address_file
  pipe_transport = address_pipe
  data = ${lookup mysql{Q_CATCHALL}}
```

Capitolo 12

Domini virtuali da LDAP

12.1 Schema LDAP

Utilizzeremo per la gestione dei domini virtuali su ldap uno schema costruito ad-hoc, chiamato mailcluster.schema (vedi appendice B). Tale schema é designato per l'uso in un mailcluster, ma alcuni suoi attributi possono essere utilizzati anche per un server singolo.
Queste sono alcune entry di esempio:

```
dn: ou=mailDomains,dc=gs,dc=net
ou: mailDomains
objectClass: organizationalUnit
objectClass: top

dn: vd=sasso.it,ou=mailDomains,dc=gs,dc=net
objectClass: mailClusterMailDomain
vd: sasso.it
description: dominio virtuale sasso.it

dn: mail=ste,vd=sasso.it,ou=mailDomains,dc=gs,dc=net
objectClass: mailClusterMailAlias
mail: ste
vd: sasso.it
description: stefano alias
mailDestination: stefano@sasso.it
mailDestination: backup@sasso.it

dn: mail=stefano,vd=sasso.it,ou=mailDomains,dc=gs,dc=net
objectClass: mailClusterMailAccount
mail: stefano
vd: sasso.it
sn: Stefano Sasso
uidNumber: 500
```

```
gidNumber: 500
mailQuota: 32M
description: stefano@sasso.it mailbox
userPassword: {CRYPT}Wo87u0MjLzdBB
mailClusterMessageStore: /srv/vmail/sasso.it/stefano/Maildir
```

12.2 Configurazione di Exim

Inseriamo all'inizio del file di configurazione

```
ldap_default_servers = <; 127.0.0.1:389 ; 192.168.171.14:389

LDAP_BINDDN = cn=admin,dc=gs,dc=net
LDAP_PASSWD = admin123
LDAP_BASEDN = dc=gs,dc=net

Q_VIRTUAL_DOMAINS = ldap:///ou=mailDomains,LDAP_BASEDN?vd?sub?\
                      (objectClass=mailClusterMailDomain)

Q_USER_ALIAS = ldap:///mail=$local_part,vd=$domain,ou=mailDomains,LDAP_BASEDN?\
                 mailDestination?sub?(objectClass=mailClusterMailAlias)

Q_MAIL_CHECK = ldap:///vd=$domain,ou=mailDomains,LDAP_BASEDN?mail?\
                 sub?(mail=$local_part)

Q_MAIL_BOX = ldap:///mail=$local_part,vd=$domain,ou=mailDomains,LDAP_BASEDN?\
               mailClusterMessageStore?sub?(objectClass=mailClusterMailAccount)

Q_MAIL_QUOTA = ldap:///mail=$local_part,vd=$domain,ou=mailDomains,LDAP_BASEDN?\
                 mailQuota?sub?(objectClass=mailClusterMailAccount)
```

poi inseriamo

```
domainlist virtual_domains = ldapm;Q_VIRTUAL_DOMAINS
```

e nelle rcpt acl:

```
  # verifica domini virtuali
  deny message = Unknown virtual user/alias
       domains = +virtual_domains
       condition = ${if eq{${lookup ldap{Q_MAIL_CHECK}{$value}fail}}{${local_part}} \
                      {no}{yes}}


  accept domains = +virtual_domains
         endpass
```

nei router

```
virtual_ldap_aliases:
  cannot_route_message = Unknown virtual alias
  driver = redirect
  domains = +virtual_domains
  data = ${lookup ldapm{Q_USER_ALIAS}{$value}fail}
  user = mail
  local_part_suffix = +*
  local_part_suffix_optional
  file_transport = address_file
  pipe_transport = address_pipe
  headers_add = X-virtual-user: $local_part\n\
                X-virtual-domain: $domain\n\
                X-virtual-transport: ldap-alias\n\
                X-virtual-address: $local_part@$domain\n\
                X-mailhub-route: $primary_hostname

virtual_ldap_mailbox:
  driver = accept
  domains = +virtual_domains
  local_part_suffix = +*
  cannot_route_message = Unknown virtual user
  local_part_suffix_optional
  condition = ${if eq{${lookup ldap{Q_MAIL_CHECK}{$value}fail}}{${local_part}}}
  transport = virtual_ldap_delivery
  headers_add = X-virtual-user: $local_part\n\
                X-virtual-domain: $domain\n\
                X-virtual-transport: ldap-maildir\n\
                X-virtual-address: $local_part@$domain\n\
                X-mailhub-route: $primary_hostname
```

e nei transport

```
virtual_ldap_delivery:
  driver = appendfile
  directory = ${lookup ldap{Q_MAIL_BOX}{$value}}
  maildir_format
  create_directory = true
  quota = ${lookup ldap{Q_MAIL_QUOTA}{$value}{5M}}
  user = mail
  group = mail
  mode = 0660
  directory_mode = 0770
```

Parte VII

Tips & Tricks

Capitolo 13

Tips & Tricks

13.1 router di sola verifica

Utilizzando *verify = recipient* è possibile creare un *router* di sola verifica, che non verrà quindi preso in considerazione durante il delivery del messaggio di posta.

```
vrfy_dest_als:
  driver = redirect
  domains = +local_domains
  data = ${lookup{$local_part}dbm \
         {/etc/exim/db/als_fwd}{$value}}
  verify_only
  verify_recipient
```

13.2 drenaggio dei messaggi

Ipotizzando di avere un cluster di mailserver, dovendo togliere un server dal cluster tutti i messaggi che sono presenti nella sua coda devono essere gestiti da un altro server... Per questo come primo router possiamo inserire un router di *drenaggio*, ovvero un router che invia tutti i messaggi ad uno specifico server.

```
drain:
  driver        = manualroute
  no_verify
  require_files = /etc/exim/db/drain_info
  route_data    = ${readfile{/etc/exim/db/drain_info}}
  transport     = smtp
```

per attivare il router basta dare quindi

```
echo mc13.mc.gs.lan > /etc/exim/db/drain_info
```

Per evitare che il server destinatario filtri di nuovo i messaggi con antivirus/antispam/quellocheè possiamo inserire un header di firma e verificarlo successivamente.

13.3 macro ACL

Durante le acl possiamo definire delle macro, chiamate acl_{m,c}{1-9} (es: acl_m7), dove le macro **m** persistono per il singolo messaggio di posta, mentre le macro **c** per la durata della connessione. Esempio:

```
warn    set acl_m7    = 1234
        condition     = ${if eq {$acl_c2}{1}{yes}{no}}
```

13.4 failing router

Possiamo inserire anche un router che fallirà sempre (ad esempio come ultimo router):

```
lists_error:
  driver  = redirect
  domains = lists.domain1344.com
  data    = :fail: \
             "${local_part}" is not a list that is managed on this system.
  allow_fail
```

13.5 scelta degli host su manualroute

Con il driver manualroute, utilizzato nei router, è possibile indicare in più modi gli host verso i quali inoltrare la posta.
Per ora abbiamo utilizzato

```
dom_relay:
  driver = manualroute
  domains = +relay_to_domains
  transport = remote_smtp
  route_data = ${lookup{$domain}lsearch{/the/file/name}}
  hosts_randomize
  no_more
```

dove, all'interno del file, era indicato

```
domain1.com: 192.168.1.3:192.168.21.5:192.168.7.5
```

È possibile, invece di specificare svariati indirizzi IP, utilizzare la risoluzione dns; sia specificando un semplice hostname

```
domain1.com: int-mx.domain1.com
```

che andando a ricercare i record MX:

```
domain1.com: int-mx.domain1.com/MX
```

13.6 maildir quota

Abbiamo già visto che per gestire la quota di una mailbox è sufficiente usare

```
quota = 10M    # (o lookup)
```

nel transport.
Tuttavia, per facilitare exim nel calcolo della quota è meglio aggiungere

```
quota_is_inclusive = false
  # fa in modo che non si tenga conto del
  # messaggio corrente per la verifica quota

maildir_tag = ,S=$message_size
quota_size_regex = ,S=(\d+)
  # aggiunge (e fa leggere) la dimensione del messaggio al nome del file
```

13.7 smtp connection rate-limiting

È possibile limitare le connessioni in ingresso:

```
#############################################################
# Ratelimiting                                              #
# We allow 1000 E-Mail pro hour for every Host. No more! #
#############################################################
  defer   message   = Sender rate exceeds $sender_rate_limit \
                      messages per $sender_rate_period
          ratelimit = 1000 / 1h / per_conn / leaky / $sender_host_address
```

13.8 tipi di lookup su file

Ci sono due modi per effettuare lookup, e numerosi tipi di file su cui farlo.
Vediamo subito un esempio dei due modi:

```
domainlist loc_domains = ${lookup{$sender_host_address}lsearch{/some/file}}
domainlist loc_domains = lsearch;/some/file
```

Nel primo caso il file deve contenere

```
192.168.3.4: domain1:domain2:...
192.168.1.9: domain3:domain4:...
```

per cui la stringa viene vista come lista di domini dopo l'espansione (quindi dopo il lookup).
A livello pratico questo fa si che il server gestisca domini diversi a seconda dell'indirizzo del sender.

Nel secondo caso vengono prese in considerazione solo le chiavi, quindi il file dovrebbe contenere:

```
domain1: bla bla bla
domain2: ble ble ble
```

In linea di massima un lookup esplicito restituisce i valori della chiave, quello implicito ritorna un elenco con solo le chiavi (e può essere usato solo come lista, quindi).

Tipi di lookup usabili (formati):

`http://www.exim.org/exim-html-4.68/doc/html/spec_html/ch09.html`

(in caso di flat file con molti record, dbm è più efficente di un lsearch)

Appendice A

Esempi di configurazioni complete

A.1 Posta per i soli utenti di sistema (senza antispam/antivirus)

Configurazione completa di un server di posta per soli utenti di sistema (no utenti virtuali), con supporto a alias e *.forward*.

A.1.1 exim4.conf

```
##########################################
# Configurazione generale                #
##########################################
primary_hostname = mailbe.azienda.it
qualify_domain = azienda.it
smtp_banner = $smtp_active_hostname ESMTP
never_users = root
hostlist   relay_from_hosts = 127.0.0.1 : aa.bb.cc.dd
domainlist local_domains = @ : localhost : azienda.it : azienda.com
domainlist relay_to_domains =
host_lookup = *
rfc1413_hosts = *
rfc1413_query_timeout = 0s
message_size_limit = 50M
return_size_limit = 100K
smtp_accept_queue = 270
smtp_accept_max = 400
smtp_accept_max_per_host = 10
smtp_accept_reserve = 100
smtp_reserve_hosts = 127.0.0.1 : ::::1 : aa.bb.cc.dd
queue_run_max = 16
ignore_bounce_errors_after = 3d
timeout_frozen_after = 3d
```

```
# definisce le acl da usare nelle varie situazioni
acl_smtp_helo = acl_check_helo
acl_smtp_rcpt = acl_check_rcpt
acl_smtp_data = acl_check_content

begin acl
##########################################
# Controllo sulla valida HELO      #
##########################################
acl_check_helo:
  # accetta se arriva da pipe locale (no tcp/ip)
  accept hosts     =:

  # accetta se arriva da un host da cui e' permesso il relay
  accept hosts     = +relay_from_hosts

  # droppa se ricevo un ip come HELO
  drop   condition = ${if match{$sender_helo_name}{^[0-9]\.[0-9]\.[0-9]\.[0-9]}{yes}{no}}
         message   = "Dropped IP-only or IP-starting helo"

  # helo non valido (RFC2821 4.1.3)
  drop   condition = ${if isip{$sender_helo_name}}
         message   = Access denied - Invalid HELO name (See RFC2821 4.1.3)
  # helo non fqdn
  drop   condition = ${if match{$sender_helo_name}{\N^\[\N}{no}{yes}}
         condition = ${if match{$sender_helo_name}{\N\.\N}{no}{yes}}
         message   = Access denied - Invalid HELO name (See RFC2821 4.1.1.1)
  drop   condition = ${if match{$sender_helo_name}{\N\.$\N}}
         message   = Access denied - Invalid HELO name (See RFC2821 4.1.1.1)
  drop   condition = ${if match{$sender_helo_name}{\N\.\.\N}}
         message   = Access denied - Invalid HELO name (See RFC2821 4.1.1.1)

  # helo e' il mio hostname
  drop   message   = "REJECTED - Bad HELO - Host impersonating [$sender_helo_name]"
         condition = ${if match{$sender_helo_name}{$primary_hostname}{yes}{no}}
  # helo e' uno dei domini gestiti da me
  drop   message   = "REJECTED - Bad HELO - Host impersonating [$sender_helo_name]"
         condition = ${if match_domain{$sender_helo_name}{+local_domains}{true}{false}}

  # rate limit, al massimo 1000 email per ora da un host
  defer  message   = Sender rate exceeds $sender_rate_limit messages \
                     per $sender_rate_period
         ratelimit = 1000 / 1h / per_conn / leaky / $sender_host_address
```

```
  accept

##############################################
# Controllo sulla validita‘ dell’RCPT     #
##############################################
acl_check_rcpt:
  # accetta se arriva da pipe locale (no tcp/ip)
  accept hosts          =:
  # nega il relay se l’indirizzo comincia con un .
  deny    local_parts   = ^.*[@%!/|] : ^\\.

  # controllo DNSBL
  drop message  = REJECTED - ${sender_host_address} is blacklisted at \
                  $dnslist_domain ($dnslist_value); ${dnslist_text}
       dnslists = sbl-xbl.spamhaus.org/<;$sender_host_address;$sender_address_domain
  drop message  = REJECTED - ${sender_address_domain} is blacklisted at \
                  ${dnslist_domain}; ${dnslist_text}
       dnslists = nomail.rhsbl.sorbs.net/$sender_address_domain
  drop message  = REJECTED - ${sender_host_address} is blacklisted at \
                  ${dnslist_domain}; ${dnslist_text}
       dnslists = zen.spamhaus.org : cbl.abuseat.org

  # i messaggi bounce da postmaster@ sono inviate solo ad un indirizzo
  drop    message       = Legitimate bounces are never sent to more than one recipient.
          senders       = : postmaster@*
          condition     = ${if >{$recipients_count}{1}{true}{false}}

  # cancella se ci sono piu‘ di 5 destinazioni fallite
  drop    message       = REJECTED - Too many failed recipients
          log_message   = REJECTED - Too many failed recipients - count = $rcpt_fail_count
          condition     = ${if > {${eval:$rcpt_fail_count}}{5}{yes}{no}}
          !verify       = recipient/callout=2m,defer_ok,use_sender

  # accetta tutte le mail per postmaster locali
  accept local_parts    = postmaster
          domains       = +local_domains

  # accetta le mail per i domini locali, dopo aver verificato il
  # recipient
  accept domains        = +local_domains
          endpass
          verify        = recipient

  # accetta se il relay e‘ consentito
  accept hosts          = +relay_from_hosts
```

```
  # non consente il resto
  deny    message       = relay not permitted

###############################################
# Controllo sulla validita' dei dati      #
###############################################
acl_check_content:

# blocca se sia il soggetto che il testo sono vuoti
  deny    message     = REJECTED - No Subject nor body
          !condition  = ${if def:h_Subject:}
          condition   = ${if <{$body_linecount}{1}{true}{false}}

  accept

begin routers
###############################################
# Domini non locali                       #
# Invia la mail al giusto MX              #
###############################################
external_gw:
  driver = dnslookup
  transport = remote_smtp
  domains = ! +local_domains
  no_more

###############################################
# Alias di sistema                        #
# Cerca un alias nel file /etc/aliases    #
###############################################
system_aliases:
  driver = redirect
  allow_fail
  allow_defer
  data = ${lookup{$local_part}lsearch{/etc/aliases}}
  user = mail
  group = mail
  local_part_suffix = +*
  local_part_suffix_optional
  headers_remove = Delivered-To
  headers_add = Delivered-To: $local_part$local_part_suffix@$domain
  headers_add = X-Mail-Suffix: $local_part_suffix
  file_transport = address_file
  pipe_transport = address_pipe
```

```
##############################################
# Forward utente                             #
# "Esegue" il file .forward nella home       #
##############################################
userforward:
  driver = redirect
  check_local_user
  file = $home/.forward
  no_verify
  no_expn
  check_ancestor
# allow_filter
  file_transport = address_file
  pipe_transport = address_pipe
  reply_transport = address_reply
  condition = ${if exists{$home/.forward}{yes}{no}}
  group = mail

##############################################
# Utente di sistema                          #
# Invia la mail nella Maildir dell'utente    #
##############################################
localuser:
  driver = accept
  check_local_user
  transport = local_delivery
  cannot_route_message = Unknown user


begin transports
##############################################
# PIPE Transport                             #
# Usato per chiamare programmi esterni       #
##############################################
address_pipe:
  driver = pipe
  return_output

##############################################
# FILE Transport                             #
# Usato per salvare su directory o file      #
##############################################
address_file:
  driver = appendfile
```

```
  delivery_date_add
  envelope_to_add
  return_path_add

##########################################
# REPLY Transport                        #
# Usato per autoreply                    #
##########################################
address_reply:
  driver = autoreply

##########################################
# SMTP Transport                         #
# Invia tramite SMTP                     #
##########################################
remote_smtp:
  driver = smtp

##########################################
# Local Delivery                         #
# Salva nella Maildir presente nella home #
##########################################
local_delivery:
  driver = appendfile
  directory_mode = 700
  group = mail
  mode = 0660
  maildir_format = true
  directory = ${home}/Maildir/
  create_directory = true
  check_string = ""
  escape_string = ""
  mode_fail_narrower = false
  envelope_to_add = true


begin retry
# This single retry rule applies to all domains and all errors. It specifies
# retries every 15 minutes for 2 hours, then increasing retry intervals,
# starting at 1 hour and increasing each time by a factor of 1.5, up to 16
# hours, then retries every 6 hours until 4 days have passed since the first
# failed delivery.
# Address or Domain     Error      Retries
# -----------------     -----      -------
*                       *          F,2h,5m; G,16h,1h,1.5; F,4d,6h
```

A.2 Antispam Gateway, con antivirus, senza delivery locale

Un gateway antispam, da usare come MX per il dominio, che effettua scansione antispam e antivirus a SMTP-time (usando *exiscan-acl*) e rigira la posta alla vera destinazione.

A.2.1 exim4.conf

```
##########################################
# Configurazione generale                #
##########################################

spamd_address = 127.0.0.1 783
av_scanner = clamd:127.0.0.1 3310

primary_hostname = mail-filter-1.company.net
qualify_domain = mail-filter-1.company.net
qualify_recipient = mail-filter-1.company.net

smtp_banner = $smtp_active_hostname ESMTP\n*** NO SPAM ALLOWED HERE ***
never_users = root
hostlist   relay_from_hosts = 127.0.0.1 : aa.bb.cc.dd
domainlist local_domains = @ : localhost : mail-filter-1.company.net
domainlist relay_to_domains = /etc/exim4/filtered-domains
host_lookup = *
rfc1413_hosts = *
rfc1413_query_timeout = 0s
message_size_limit = 50M
return_size_limit = 100K
smtp_accept_queue = 1500
smtp_accept_max = 1500
smtp_accept_max_per_host = 1500
smtp_accept_queue_per_connection = 1500
smtp_accept_reserve = 100
smtp_reserve_hosts = 127.0.0.1 : ::::1 : aa.bb.cc.dd
queue_run_max = 32
ignore_bounce_errors_after = 3d
timeout_frozen_after = 3d
# custom queue/delivery options
remote_max_parallel = 8

#definisce le acl da usare nelle varie situazioni
acl_smtp_helo = acl_check_helo
acl_smtp_rcpt = acl_check_rcpt
acl_smtp_data = acl_check_content
```

```
acl_check_helo:
  # accetta se arriva da pipe locale (no tcp/ip)
  accept hosts     =:
  # accetta se arriva da un host da cui e‘ permesso il relay
  accept hosts     = +relay_from_hosts

  # droppa se ricevo un ip come HELO
  drop   condition = ${if match{$sender_helo_name}{^[0-9]\.[0-9]\.[0-9]\.[0-9]}{yes}{no}}
         message   = "Dropped IP-only or IP-starting helo"
  # helo non valido (RFC2821 4.1.3)
  drop   condition = ${if isip{$sender_helo_name}}
         message   = Access denied - Invalid HELO name (See RFC2821 4.1.3)
  # helo non fqdn
  drop   condition = ${if match{$sender_helo_name}{\N^\[\N}{no}{yes}}
         condition = ${if match{$sender_helo_name}{\N\.\N}{no}{yes}}
         message   = Access denied - Invalid HELO name (See RFC2821 4.1.1.1)
  drop   condition = ${if match{$sender_helo_name}{\N\.$\N}}
         message   = Access denied - Invalid HELO name (See RFC2821 4.1.1.1)
  drop   condition = ${if match{$sender_helo_name}{\N\.\.\N}}
         message   = Access denied - Invalid HELO name (See RFC2821 4.1.1.1)
  # helo e‘ il mio hostname
  drop   message   = "REJECTED - Bad HELO - Host impersonating [$sender_helo_name]"
         condition = ${if match{$sender_helo_name}{$primary_hostname}{yes}{no}}
  # helo e‘ uno dei domini gestiti da me
  drop   message   = "REJECTED - Bad HELO - Host impersonating [$sender_helo_name]"
         condition = ${if match_domain{$sender_helo_name}{+local_domains}{true}{false}}

  # rate limit, al massimo 1000 email per ora da un host
  defer message    = Sender rate exceeds $sender_rate_limit messages \
                     per $sender_rate_period
       ratelimit = 1000 / 1h / per_conn / leaky / $sender_host_address

  accept

##############################################
# Controllo sulla validita‘ dell’RCPT     #
##############################################
acl_check_rcpt:
  # accetta se arriva da pipe locale (no tcp/ip)
  accept hosts          =:
  # nega il relay se l’indirizzo comincia con un .
  deny    local_parts   = ^.*[@%!/|] : ^\\.

  # dnsbl
  drop message  = REJECTED - ${sender_host_address} is blacklisted at \
```

```
                $dnslist_domain ($dnslist_value); ${dnslist_text}
     dnslists = sbl-xbl.spamhaus.org/<;$sender_host_address;$sender_address_domain
  drop message  = REJECTED - ${sender_address_domain} is blacklisted at \
                $dnslist_domain}; ${dnslist_text}
     dnslists = nomail.rhsbl.sorbs.net/$sender_address_domain
  drop message  = REJECTED - ${sender_host_address} is blacklisted at \
                $dnslist_domain}; ${dnslist_text}
     dnslists = zen.spamhaus.org : cbl.abuseat.org

  # i messaggi bounce da postmaster@ sono inviate solo ad un indirizzo
  drop    message       = Legitimate bounces are never sent to more than one recipient.
          senders       = : postmaster@*
          condition     = ${if >{$recipients_count}{1}{true}{false}}

  # cancella se ci sono piu' di 5 destinazioni fallite
  drop    message       = REJECTED - Too many failed recipients
          log_message   = REJECTED - Too many failed recipients - count = $rcpt_fail_count
          condition     = ${if > {${eval:$rcpt_fail_count}}{5}{yes}{no}}
          !verify       = recipient/callout=2m,defer_ok,use_sender

  # accetta tutte le mail per postmaster locali
  accept local_parts    = postmaster
          domains       = +local_domains
  # accetta le mail per i domini locali, dopo aver verificato il
  # recipient
  accept domains        = +local_domains
          endpass
          verify        = recipient

  # accetta il relay to host, verificando il recipient (e il sender)
  accept domains        = +relay_to_domains
         endpass
         message        = cannot verify sender
         verify         = sender/no_details/defer_ok
         message        = relay to $local_part@$domain not allowed
         verify         = recipient/no_details/callout=use_postmaster,defer_ok

  # accetta se il relay e' consentito
  accept hosts          = +relay_from_hosts

  # non consente il resto
  deny    message       = relay not permitted

##########################################
# Controllo sulla validita' dei dati     #
```

```
##############################################
acl_check_content:
  # blocca se sia il soggetto che il testo sono vuoti
  deny    message       = REJECTED - No Subject nor body
          !condition    = ${if def:h_Subject:}
          condition     = ${if <{$body_linecount}{1}{true}{false}}

  deny    message       = This message contains malformed MIME ($demime_reason)
          demime        = *
          condition     = ${if >{$demime_errorlevel}{2}}

  deny    message       = Mail contains blacklisted attachment (.$found_extension)
          demime        = bat:com:exe:pif:prf:scr:vbs

  warn    message       = X-Spam-Score: $spam_score ($spam_bar)
          spam          = Debian-exim:true
  warn    message       = X-Spam-Report: $spam_report
          spam          = Debian-exim:true
  #warn    message       = Subject: **** SPAM **** $h_Subject
  #        spam          = Debian-exim
  warn    message       = X-Spam-Flag: YES
          spam          = Debian-exim:true
          condition     = ${if >{$spam_score_int}{50}}
  deny    message       = This message scored $spam_score spam points.
          spam          = Debian-exim:true
          condition     = ${if >{$spam_score_int}{100}}

  deny    message       = This message contains a virus or other harmful content ($malware_name)
          malware       = *

  accept

begin routers
redirect:
  driver = redirect
  data = ${lookup{$local_part@$domain}lsearch{/etc/exim4/mail-redirect}}

internal:
  driver = manualroute
  domains = +relay_to_domains
  transport = remote_smtp
  route_data = ${lookup{$domain}partial-lsearch{/etc/exim4/remote-servers}}
  no_more

external_gw:
```

```
  driver = dnslookup
  transport = remote_smtp
  domains = ! +local_domains
  no_more

system_aliases:
  driver = redirect
  allow_fail
  allow_defer
  data = ${lookup{$local_part}lsearch{/etc/aliases}}
  user = mail
  group = mail
  local_part_suffix = +*
  local_part_suffix_optional
  file_transport = address_file
  pipe_transport = address_pipe


begin transports
##########################################
# PIPE Transport                         #
# Usato per chiamare programmi esterni   #
##########################################
address_pipe:
  driver = pipe
  return_output
##########################################
# FILE Transport                         #
# Usato per salvare su directory o file  #
##########################################
address_file:
  driver = appendfile
  delivery_date_add
  envelope_to_add
  return_path_add

##########################################
# SMTP Transport                         #
# Invia tramite SMTP                     #
##########################################
remote_smtp:
  driver = smtp


begin retry
```

```
# This single retry rule applies to all domains and all errors. It specifies
# retries every 15 minutes for 2 hours, then increasing retry intervals,
# starting at 1 hour and increasing each time by a factor of 1.5, up to 16
# hours, then retries every 6 hours until 4 days have passed since the first
# failed delivery.
# Address or Domain     Error      Retries
# -----------------     -----      -------
*                       *          F,5h,5m; G,16h,1h,1.5; F,4d,6h
```

A.2.2 filtered-domains

```
dominio1.net
dominio2.com
```

A.2.3 remote-servers

```
dominio1.net: mail-be-1.mail.azienda.com:mail-be-2.mail.azienda.com
dominio2.com: mail-interno.dominio2.com
```

A.2.4 mail-redirect

```
postmaster@dominio2.com: amministratore@dominio3.org
```

A.3 Mail Relay con autenticazione SMTP

Configurazione completa di un server per la posta in uscita con autenticazione SMTP.

A.3.1 exim4.conf

```
# carica funzioni perl
perl_startup = do '/etc/exim4/perl_exim4.pl'
perl_at_start

primary_hostname = mailout.mail.dominio.it
qualify_domain = mailout.mail.dominio.it
smtp_banner = $smtp_active_hostname ESMTP
never_users = root
hostlist   relay_from_hosts = 127.0.0.1 : aa.bb.cc.dd : aa.bb.cc.ee
domainlist local_domains = @ : localhost : mailout.mail.dominio.it
domainlist relay_to_domains =
host_lookup = *
rfc1413_hosts = *
rfc1413_query_timeout = 0s
```

```
message_size_limit = 50M
return_size_limit = 100K
smtp_accept_queue = 1500
smtp_accept_max = 1500
smtp_accept_max_per_host = 1500
smtp_accept_queue_per_connection = 1500
smtp_accept_reserve = 100
smtp_reserve_hosts = 127.0.0.1 : ::::1 : aa.bb.cc.dd : aa.bb.cc.ee
queue_run_max = 32
ignore_bounce_errors_after = 3d
timeout_frozen_after = 3d

# custom queue/delivery options
remote_max_parallel = 8

tls_advertise_hosts = *
tls_certificate = /etc/exim4/exim4.pem
tls_privatekey = /etc/exim4/exim4.pem
daemon_smtp_ports = 25 : 465 : 587
tls_on_connect_ports = 465


#definisce le acl da usare nelle varie situazioni
acl_smtp_helo = acl_check_helo
acl_smtp_rcpt = acl_check_rcpt
acl_smtp_data = acl_check_content




begin acl
##########################################
# Controllo sulla valida HELO            #
##########################################
acl_check_helo:
  # accetta se arriva da pipe locale (no tcp/ip)
  accept hosts     =:
  # accetta se arriva da un host da cui e‘ permesso il relay
  accept hosts     = +relay_from_hosts

  # rate limit, al massimo 1000 email per ora da un host
  defer message   = Sender rate exceeds $sender_rate_limit messages \
                    per $sender_rate_period
        ratelimit = 1500 / 1h / per_conn / leaky / $sender_host_address
```

```
  accept

##############################################
# Controllo sulla validita‘ dell’RCPT     #
##############################################
acl_check_rcpt:
  # accetta se arriva da pipe locale (no tcp/ip)
  accept hosts          =:
  # nega il relay se l’indirizzo comincia con un .
  deny    local_parts   = ^.*[@%!/|] : ^\\.

  # accetta da sorgenti autenticate
  accept authenticated = *
         control       = submission

  # controlli normali
  # droppa se ricevo un ip come HELO
  drop   condition = ${if match{$sender_helo_name}{^[0-9]\.[0-9]\.[0-9]\.[0-9]}{yes}{no}}
         message   = "Dropped IP-only or IP-starting helo"
  # helo non valido (RFC2821 4.1.3)
  drop   condition = ${if isip{$sender_helo_name}}
         message   = Access denied - Invalid HELO name (See RFC2821 4.1.3)
  # helo non fqdn
  drop   condition = ${if match{$sender_helo_name}{\N^\[\N}{no}{yes}}
         condition = ${if match{$sender_helo_name}{\N\.\N}{no}{yes}}
         message   = Access denied - Invalid HELO name (See RFC2821 4.1.1.1)
  drop   condition = ${if match{$sender_helo_name}{\N\.$\N}}
         message   = Access denied - Invalid HELO name (See RFC2821 4.1.1.1)
  drop   condition = ${if match{$sender_helo_name}{\N\.\.\N}}
         message   = Access denied - Invalid HELO name (See RFC2821 4.1.1.1)
  # helo e‘ il mio hostname
  drop   message   = "REJECTED - Bad HELO - Host impersonating [$sender_helo_name]"
         condition = ${if match{$sender_helo_name}{$primary_hostname}{yes}{no}}
  # helo e‘ uno dei domini gestiti da me
  drop   message   = "REJECTED - Bad HELO - Host impersonating [$sender_helo_name]"
         condition = ${if match_domain{$sender_helo_name}{+local_domains}{true}{false}}

  # DNSBL
  drop message  = REJECTED - ${sender_host_address} is blacklisted at \
                  $dnslist_domain ($dnslist_value); ${dnslist_text}
       dnslists = sbl-xbl.spamhaus.org/<;$sender_host_address;$sender_address_domain
  drop message  = REJECTED - ${sender_address_domain} is blacklisted at \
                  ${dnslist_domain}; ${dnslist_text}
       dnslists = nomail.rhsbl.sorbs.net/$sender_address_domain
  drop message  = REJECTED - ${sender_host_address} is blacklisted at \
```

```
                ${dnslist_domain}; ${dnslist_text}
      dnslists = zen.spamhaus.org : cbl.abuseat.org

  # i messaggi bounce da postmaster@ sono inviate solo ad un indirizzo
  drop    message       = Legitimate bounces are never sent to more than one recipient.
          senders       = : postmaster@*
          condition     = ${if >{$recipients_count}{1}{true}{false}}

   # accetta tutte le mail per postmaster locali
   accept local_parts    = postmaster
           domains       = +local_domains
   # accetta le mail per i domini locali, dopo aver verificato il
   # recipient
   accept domains        = +local_domains
           endpass
           verify        = recipient

  # accetta se il relay e' consentito
   accept hosts          = +relay_from_hosts

   # non consente il resto
   deny    message       = relay not permitted

##############################################
# Controllo sulla validita' dei dati      #
##############################################
acl_check_content:

  # blocca se sia il soggetto che il testo sono vuoti
  deny    message     = REJECTED - No Subject nor body
          !condition  = ${if def:h_Subject:}
          condition   = ${if <{$body_linecount}{1}{true}{false}}

  accept

begin routers
##############################################
# Domini non locali                       #
# Invia la mail al giusto MX              #
##############################################
external_gw:
  driver = dnslookup
  transport = remote_smtp
  domains = ! +local_domains
  no_more
```

```
##############################################
# Alias di sistema                           #
# Cerca un alias nel file /etc/aliases       #
##############################################
system_aliases:
  driver = redirect
  allow_fail
  allow_defer
  data = ${lookup{$local_part}lsearch{/etc/aliases}}
  user = mail
  group = mail
  local_part_suffix = +*
  local_part_suffix_optional
  headers_remove = Delivered-To
  headers_add = Delivered-To: $local_part$local_part_suffix@$domain
  file_transport = address_file
  pipe_transport = address_pipe

begin transports
##############################################
# PIPE Transport                             #
# Usato per chiamare programmi esterni       #
##############################################
address_pipe:
  driver = pipe
  return_output
##############################################
# FILE Transport                             #
# Usato per salvare su directory o file      #
##############################################
address_file:
  driver = appendfile
  delivery_date_add
  envelope_to_add
  return_path_add
##############################################
# SMTP Transport                             #
# Invia tramite SMTP                         #
##############################################
remote_smtp:
  driver = smtp

begin retry
# This single retry rule applies to all domains and all errors. It specifies
```

```
# retries every 15 minutes for 2 hours, then increasing retry intervals,
# starting at 1 hour and increasing each time by a factor of 1.5, up to 16
# hours, then retries every 6 hours until 4 days have passed since the first
# failed delivery.
# Address or Domain     Error      Retries
# -----------------     -----      -------
*                       *          F,5h,5m; G,16h,1h,1.5; F,4d,6h

begin rewrite
  *@+local_domains "${lookup{${local_part}}lsearch{/etc/exim4/email-addresses}{$value}fail}" Ffrs


begin authenticators
plain:
  driver           = plaintext
  public_name      = PLAIN
  server_condition = ${perl{smtplogin}{/etc/exim4/passwd.smtpd}{$2}{$3}}

login:
  driver           = plaintext
  public_name      = LOGIN
  server_prompts   = "Username:: : Password::"
  server_condition = ${perl{smtplogin}{/etc/exim4/passwd.smtpd}{$1}{$2}}
```

A.3.2 perl_exim4.pl

```
#!/usr/bin/perl

use Apache::Htpasswd;

sub smtplogin
{
  my $file = shift;
  my $account = shift;
  my $password = shift;
  if ( ! -r $file )
  {
    return 0;
  }
  $b = new Apache::Htpasswd({passwdFile => $file,
     ReadOnly   => 1});
  if ($b->htCheckPassword($account, $password))
  {
    return 1;
  }
```

```
  else
  {
    return 0;
  }
}
```

A.4 Domini virtuali su MySQL senza antispam/antivirus

Configurazione completa di un server di posta per la gestione virtuale di domini, utenti e alias basata su MySQL. Il database MySQL é predisposto per salvare il tipo di controllo antivirus/antispam definito dall'utente, ma non verrá usato in questa configurazione.

A.4.1 Database MySQL

```
CREATE TABLE aliases (
  aliasid bigint(20) NOT NULL auto_increment,
  aliasname varchar(100) NOT NULL,
  domain varchar(100) NOT NULL,
  destination varchar(201) NOT NULL,
  active tinyint(1) NOT NULL default '1',
  PRIMARY KEY  (aliasid)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;

CREATE TABLE domains (
  id int(11) NOT NULL auto_increment,
  domain varchar(100) NOT NULL,
  adminpassword varchar(100) NOT NULL,
  catchall varchar(201),
  created datetime NOT NULL,
  active tinyint(1) NOT NULL default '1',
  PRIMARY KEY  (id),
  UNIQUE KEY domain (domain)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;

CREATE TABLE users (
  userid bigint(20) NOT NULL auto_increment,
  username varchar(100) NOT NULL,
  domain varchar(100) NOT NULL,
  email varchar(201) NOT NULL,
  password varchar(200) NOT NULL,
  maildir varchar(255) NOT NULL,
  active tinyint(1) NOT NULL default '1',
  vacation tinyint(1) NOT NULL default '0',
  vacationsubj varchar(250) NOT NULL,
```

```
  vacationmsg text NOT NULL,
  uid int(11) NOT NULL,
  gid int(11) NOT NULL,
  quota varchar(50) NOT NULL,
  spamScanType set('INACTIVE','SPAMFOLDER','MARK','SUBJECT') NOT NULL default 'SPAMFOLDER',
  virusScanType set('INACTIVE','SPAMFOLDER','REJECT','SUBJECT') NOT NULL default 'REJECT',
  created datetime NOT NULL,
  PRIMARY KEY  (userid)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;

CREATE TABLE exim_greylist
(
  id int(11) NOT NULL auto_increment PRIMARY KEY,
  relay_ip varchar(21),
  from_domain varchar(85),
  block_expires datetime NOT NULL,
  record_expires datetime NOT NULL,
  origin_type enum('MANUAL','AUTO') NOT NULL DEFAULT 'AUTO',
  create_time datetime NOT NULL,
  KEY exim_lookup (relay_ip,from_domain)
);
```

A.4.2 exim4.conf

```
hide mysql_servers = "127.0.0.1/exim/exim/exim"
Q_LOCAL_DOMAIN=SELECT domain FROM domains WHERE domain=\'$domain\'
Q_MAIL_ALIAS=SELECT destination FROM aliases WHERE aliasname=\'$local_part\' AND domain=\'$domain
Q_VALID_EMAIL=SELECT userid FROM users WHERE email=\'$local_part@$domain\'

Q_COUNT_MAIL=SELECT COUNT(*) FROM users WHERE email=\'$local_part@$domain\'
Q_COUNT_ALIAS=SELECT COUNT(*) FROM aliases WHERE aliasname=\'$local_part\' AND domain=\'$domain\'

Q_MAIL_BOX_DIR=SELECT maildir FROM users WHERE email=\'$local_part@$domain\'
Q_MAIL_BOX_QUOTA=SELECT quota FROM users WHERE email=\'$local_part@$domain\'

Q_ISAWAY=SELECT domain FROM users WHERE domain='${quote_mysql:$domain}' AND username='${quote_mys
Q_AWAYTEXT=SELECT vacationmsg FROM users WHERE domain='${quote_mysql:$domain}' AND username='${qu

Q_CATCHALL=SELECT adminpassword FROM domains WHERE domain=\'$domain\'

GREYLIST_TEST = SELECT IF(NOW() > block_expires, 2, 1) \
  FROM exim_greylist \
  WHERE relay_ip = '${quote_mysql:$sender_host_address}' \
  AND from_domain = '${quote_mysql:$sender_address_domain}' \
  AND record_expires > NOW()
```

```
GREYLIST_ADD = \
  INSERT INTO exim_greylist \
  SET relay_ip   = '${quote_mysql:$sender_host_address}', \
  from_domain    = '${quote_mysql:$sender_address_domain}', \
  block_expires = DATE_ADD(NOW(), INTERVAL 3 MINUTE), \
  record_expires = DATE_ADD(NOW(), INTERVAL 28 DAY), \
  origin_type    = 'AUTO', \
  create_time    = NOW()
GREYLIST_UPDATE = \
  UPDATE exim_greylist \
  SET    record_expires = DATE_ADD(now(), INTERVAL 28 DAY) \
  WHERE relay_ip        = '${quote_mysql:$sender_host_address}' \
  AND    from_domain    = '${quote_mysql:$sender_address_domain}' \
  AND    record_expires > NOW()
```

```
##########################################
# Configurazione generale                #
##########################################
#primary_hostname = simplemx.dominio.net
qualify_domain = dominio.net
smtp_banner = $smtp_active_hostname ESMTP Exim\n$tod_full\nHi spammer!
never_users = root
hostlist   relay_from_hosts = 127.0.0.1 : 72.20.214.0/24


domainlist local_domains = @ : dominio.net : posta.dominio.net : lsearch;/etc/exim4/vdomains
domainlist virtual_domains = mysql;Q_LOCAL_DOMAIN
domainlist my_domains = +local_domains : +virtual_domains

#host_lookup = *
rfc1413_hosts = *
rfc1413_query_timeout = 0s
message_size_limit = 50M
return_size_limit = 100K
smtp_accept_queue = 270
smtp_accept_max = 400
smtp_accept_max_per_host = 10
smtp_accept_reserve = 100
smtp_reserve_hosts = 127.0.0.1 : ::::1 : 72.20.214.0/24
queue_run_max = 16
ignore_bounce_errors_after = 3d
```

```
timeout_frozen_after = 3d

#definisce le acl da usare nelle varie situazioni

acl_smtp_helo = acl_check_helo
acl_smtp_rcpt = acl_check_rcpt
acl_smtp_data = acl_check_content

begin acl

##########################################
# Controllo sulla validità dell'HELO     #
##########################################
acl_check_helo:
  # accetta se arriva da pipe locale (no tcp/ip)
  accept hosts     = :
  # accetta se arriva da un host da cui è permesso il relay
  accept hosts     = +relay_from_hosts
  # droppa se ricevo come HELO il mio ip
  drop   condition = ${if match{$sender_helo_name}{MY_IP}{yes}{no} }
         message   = "Dropped spammer pretending to be us"
  # droppa se ricevo un ip come HELO
  drop   condition = ${if match{$sender_helo_name}\
                       {^[0-9]\.[0-9]\.[0-9]\.[0-9]}{yes}{no}}
         message   = "Dropped IP-only or IP-starting helo"
  accept

##########################################
# Controllo sulla validità dell'RCPT     #
##########################################
acl_check_rcpt:
  # accetta se arriva da pipe locale (no tcp/ip)
  accept  hosts         = :
  # nega il relay se l'indirizzo comincia con un .
  deny    local_parts   = ^.*[@%!/|] : ^\\.

  warn
    set acl_m2      = ${lookup mysql{GREYLIST_TEST}{$value}{0}}
    defer
#   ! hosts         = +whitelist
    ! hosts         = +relay_from_hosts
    ! authenticated = *
    condition       = ${if eq{$acl_m2}{0}{yes}}
    condition       = ${lookup mysql{GREYLIST_ADD}{yes}{no}}
    message         = Now greylisted - please try again in five minutes.
```

```
  defer
#   ! hosts         = +whitelist
    ! hosts         = +relay_from_hosts
    ! authenticated = *
    condition       = ${if eq{$acl_m2}{1}{yes}}
    message         = Still greylisted - please try again in five minutes.
  defer
#   ! hosts         = +whitelist
    ! hosts         = +relay_from_hosts
    ! authenticated = *
    condition       = ${lookup mysql{GREYLIST_UPDATE}{no}{no}}
    message         = Greylist update failed


  drop message = REJECTED - too many failed rcpt count = $rcpt_fail_count
       log_message = rejected: too many failed rcpt count = $rcpt_fail_count
       condition = ${if > {${eval:$rcpt_fail_count}}{5}{yes}{no}}


  # accetta tutte le mail per postmaster locali
  accept  local_parts   = postmaster
          domains       = +my_domains

  # verifica domini virtuali
  deny message = Unknown virtual user/alias
       domains = +virtual_domains
       condition = ${if and {{eq {${lookup mysql{Q_COUNT_MAIL}}}{0}}\
         {eq {${lookup mysql{Q_COUNT_ALIAS}}}{0}}}}


  accept domains = +virtual_domains
         endpass


  # accetta le mail per i domini locali, dopo aver verificato il
  # recipient
  accept  domains       = +local_domains
          endpass
          verify        = recipient
  # accetta se il relay è consentito
  accept  hosts         = +relay_from_hosts
  # non consente il resto
  deny    message       = relay not permitted

##############################################
```

```
# Controllo sulla validità dei dati       #
#############################################
acl_check_content:
  accept

begin routers

posta_fail:
  driver = redirect
  domains = posta.dominio.net
  condition = ${if eq{$local_part}{postmaster}{no}{yes}}
  allow_fail
  data = :fail: posta.dominio.net accepts only postmaster, no "${local_part}"
  no_more

virtual_domain_aliases:
  driver = redirect
  domains = lsearch;/etc/exim4/vdomains
  data = ${lookup{$local_part}lsearch{/etc/exim4/aliases-$domain}}
  headers_add = X-virtual-user: $local_part\n\
                X-virtual-domain: $domain\n\
                X-virtual-address: $local_part@$domain\n\
                X-mailhub-route: $primary_hostname
  no_more
  cannot_route_message = Unknown vdomain text alias


virtual_mysql_aliases:
  cannot_route_message = Unknown virtual alias
  driver = redirect
  domains = +virtual_domains
  data = ${lookup mysql{Q_MAIL_ALIAS}{$value}}
#  rewrite = true
  user = mail
  local_part_suffix = +*
  local_part_suffix_optional
  file_transport = address_file
  pipe_transport = address_pipe
  headers_add = X-virtual-user: $local_part\n\
                X-virtual-domain: $domain\n\
                X-virtual-transport: mysql-alias\n\
                X-virtual-address: $local_part@$domain\n\
                X-mailhub-route: $primary_hostname
```

```
vacation_router:
  driver = accept
  domains = ${lookup mysql {Q_ISAWAY}{$value}}
  transport = vacation_autoreply
  unseen


virtual_mysql_mailbox:
  driver = accept
  domains = +virtual_domains
  local_part_suffix = +*
  cannot_route_message = Unknown mysql virtual user
  local_part_suffix_optional
  condition = ${lookup mysql{Q_VALID_EMAIL}}
  transport = virtual_mysql_delivery
  headers_add = X-virtual-user: $local_part\n\
                X-virtual-domain: $domain\n\
                X-virtual-transport: mysql-maildir\n\
                X-virtual-address: $local_part@$domain\n\
                X-mailhub-route: $primary_hostname


mysql_catchall:
  driver = redirect
  domains = +virtual_domains
  file_transport = address_file
  pipe_transport = address_pipe
  data = ${lookup mysql{Q_CATCHALL}{$value}}


##########################################
# Domini non locali                      #
# Invia la mail al giusto MX             #
##########################################
external_gw:
  driver = dnslookup
  transport = remote_smtp
  domains = ! +my_domains
  no_more

##########################################
# Alias di sistema                       #
# Cerca un alias nel file /etc/aliases   #
##########################################
```

```
system_aliases:
  driver = redirect
  domains = +local_domains
  allow_fail
  allow_defer
  data = ${lookup{$local_part}lsearch{/etc/aliases}}
  user = mail
  group = mail
  file_transport = address_file
  pipe_transport = address_pipe

###############################################
# Forward utente                              #
# "Esegue" il file .forward nella home        #
###############################################
userforward:
  driver = redirect
  domains = +local_domains
  check_local_user
  file = $home/.forward
  no_verify
  no_expn
  check_ancestor
# allow_filter
  file_transport = address_file
  pipe_transport = address_pipe
  reply_transport = address_reply
  condition = ${if exists{$home/.forward} {yes} {no} }
  group = mail

###############################################
# Utente di sistema                           #
# Invia la mail nella Maildir dell'utente     #
###############################################
localuser:
  driver = accept
  check_local_user
  transport = mail_spool
  domains = +local_domains
  cannot_route_message = Unknown local user


begin transports

vacation_autoreply:
```

```
  driver = autoreply
  to = ${sender_address}
  from = ${local_part}@${domain}
  subject = Vacation for ${local_part}@${domain}
  text = ${lookup mysql {Q_AWAYTEXT}{$value}}

virtual_mysql_delivery:
  driver = appendfile
  directory = /srv/vmail/${lookup mysql{Q_MAIL_BOX_DIR}{$value}}
  maildir_format
  create_directory = true
  quota = ${lookup mysql{Q_MAIL_BOX_QUOTA}{$value}{5M}}
  user = mail
  group = mail
  mode = 0660
  directory_mode = 0770




##############################################
# PIPE Transport                             #
# Usato per chiamare programmi esterni       #
##############################################
address_pipe:
  driver = pipe
  return_output
  return_path_add

##############################################
# FILE Transport                             #
# Usato per salvare su directory o file      #
##############################################
address_file:
  driver = appendfile
  delivery_date_add
  envelope_to_add
  return_path_add

##############################################
# REPLY Transport                            #
# Usato per autoreply                        #
##############################################
address_reply:
  driver = autoreply
```

```
##########################################
# SMTP Transport                         #
# Invia tramite SMTP                     #
##########################################
remote_smtp:
  driver = smtp

##########################################
# Local Delivery                         #
# Salva nella Maildir presente nella home #
##########################################
local_delivery:
  driver = appendfile
  create_directory = true
  directory_mode = 700
  group = mail
  mode = 0660
  maildir_format = true
  directory = ${home}/Maildir/
  check_string = ""
  escape_string = ""
  mode_fail_narrower = false
  envelope_to_add = true

mail_spool:
  debug_print = "T: appendfile for $local_part@$domain"
  driver = appendfile
  file = /var/mail/$local_part
  delivery_date_add
  envelope_to_add
  return_path_add
  group = mail
  mode = 0660
  mode_fail_narrower = false


begin retry

# This single retry rule applies to all domains and all errors. It specifies
# retries every 15 minutes for 2 hours, then increasing retry intervals,
# starting at 1 hour and increasing each time by a factor of 1.5, up to 16
# hours, then retries every 6 hours until 4 days have passed since the first
# failed delivery.
```

```
# Address or Domain    Error       Retries
# ------------------    -----       -------

*                      *           F,2h,5m; G,16h,1h,1.5; F,4d,6h
```

Appendice B

mailcluster.schema

```
#mailcluster.schema

objectIdentifier mailClusterOID 1.3.6.1.4.1.29283
objectIdentifier mailClusterLDAP mailClusterOID:2
objectIdentifier mailClusterAttributeType mailClusterLDAP:1
objectIdentifier mailClusterObjectClass mailClusterLDAP:2


objectIdentifier String 1.3.6.1.4.1.1466.115.121.1.26
objectIdentifier Boolean 1.3.6.1.4.1.1466.115.121.1.7
objectIdentifier Date 1.3.6.1.4.1.1466.115.121.1.26
objectIdentifier Counter 1.3.6.1.4.1.1466.115.121.1.27



attributetype ( mailClusterAttributeType:1 NAME 'mailClusterNode'
DESC 'Mail Cluster Node Where to store mails'
EQUALITY caseExactIA5Match
SYNTAX String SINGLE-VALUE )

attributetype ( mailClusterAttributeType:2 NAME 'mailClusterMessageStore'
DESC 'Folder Where to store mail'
EQUALITY caseExactIA5Match
SYNTAX String SINGLE-VALUE )

attributetype ( mailClusterAttributeType:3 NAME 'mailQuota'
DESC 'Maildir quota'
EQUALITY caseExactIA5Match
SYNTAX String SINGLE-VALUE )

attributetype ( mailClusterAttributeType:4 NAME 'vd'
```

```
DESC 'Mail Virtual Domain'
EQUALITY caseIgnoreIA5Match
SUBSTR caseIgnoreIA5SubstringsMatch
SYNTAX String )

attributetype ( mailClusterAttributeType:5 NAME 'mailDestination'
DESC 'Mailbox alias'
EQUALITY caseIgnoreIA5Match
SUBSTR caseIgnoreIA5SubstringsMatch
SYNTAX String )

objectclass ( mailClusterObjectClass:1 NAME 'mailClusterMailAccount'
SUP top STRUCTURAL
DESC 'Mail Account'
MUST ( mail $ userPassword $ mailClusterMessageStore )
MAY ( vd $ uidNumber $ gidNumber $ sn $ mailQuota $ mailClusterNode $ description ) )

objectclass ( mailClusterObjectClass:2 NAME 'mailClusterMailAlias'
SUP top STRUCTURAL
DESC 'Mail Alias'
MUST ( mail $ mailDestination )
MAY ( vd $ description ) )

objectclass ( mailClusterObjectClass:3 NAME 'mailClusterMailDomain'
SUP top STRUCTURAL
DESC 'Mail Domain'
MUST ( vd )
MAY ( description ) )
```