

Insider Tips: Email Clusters

The Fastest Postman

Providing basic email facilities is a major challenge to admins in large-scale networks. Most Linux distributions do their best to forward electronic mail, but real admins can do better. Setting up Mail Transfer Agents need not be a nightmare task if you spend a little time planning what you want to achieve. Multiple servers can be used to share the busy load in many different ways, ensuring the mail always gets through. **BY MARC ANDRÉ SELIG**

Users of stand-alone workstations are unlikely to have trouble with email. Mail User Agents (MUAs) such as Mozilla, KMail or Evolution pick up your mail from your Internet provider and send new messages to that provider. In other words, you talk directly to your provider's Mail Transfer Agent, as shown in Figure 1.

This configuration is designed to handle a single user with a single program. If the user has multiple clients, she will normally be required to set them up separately. In the case of a workstation that supports multiple users, it makes sense to set up a local Mail Transfer Agent (MTA) (see Figure 2). Sendmail [1] is the heritage MTA, although Qmail [2], and increasingly Postfix [3] are also popular. The MTA allows you to set up automatic processes to dispatch email messages. This job is typically handled by the cron daemon, which will send any output from its jobs to root.

Unscalable

Unfortunately, this setup has limitations, even in a small-scale server facility with some 30 or 40 devices. Imagine each host wanting to dispatch its own messages. It would be more or less impossible to keep track. And you would need to re-configure every single host on your network if the boss wanted to see some statistics, or if you needed to install encryption facilities or a standardized archive. Obviously, you will want to avoid all that work, and



smart hosts (see Figure 3). These systems accept messages sent by the internal systems and forward them to the recipient, no matter if the final destination is on, or outside of, the local network. You can offload advanced email functionality to the smart host. And analyzing the logfiles on this system will provide those outgoing mail statistics your boss asked for.

Internal hosts simply forward any messages to the same relay. Listing 1 shows the smart host configuration for Sendmail in the M4 file `.../sendmail/cf/cf/sendmail.mc`. At the same time, this configuration disables the Message Submission Agent, which is seldom required, and restricts the daemon to local SMTP (Simple Mail Transfer Protocol, [5]) connections. The smart host setup for Qmail is far simpler. A single line in `/var/qmail/control/smtproutes` is all you need:

```
:relay.myorg.uk
```

The Qmail SMTP daemon is called in `inetd`. And this is the only place to restrict the daemon to localhost – by using TCP Wrapper for example. Postfix allows you to include both these settings in `/etc/postfix/main.cf`:

```
relayhost = relay.myorg.uk
inet_interfaces = 127.0.0.1
```

opt for a centralized mail infrastructure instead. Also, the task of packet filtering is far easier if you prevent internal devices from contacting the outside world directly.

United We Stand

Most large-scale installations designate one or more computers as relays or

Of course, you will need to ensure that the central relay provides this service only to internal systems to avoid spam-

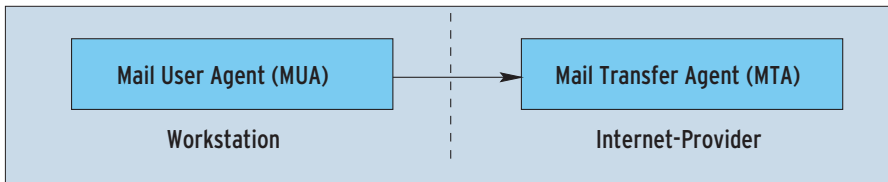


Figure 1: Email handling is simple for stand-alone systems. Your mail client (MUA) contacts the provider's mail server (MTA) directly

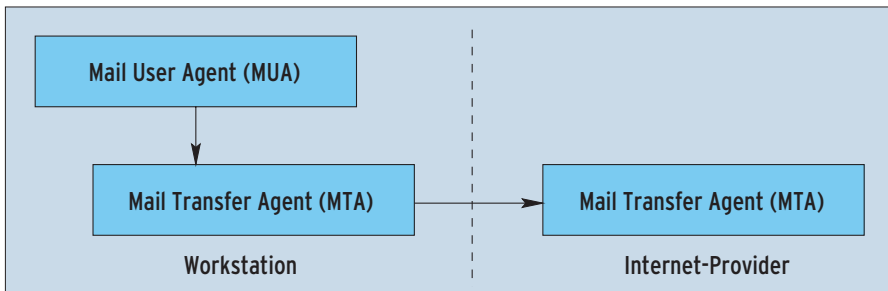


Figure 2: When a message is dispatched by a stand-alone workstation, the MUA hands the message to the local MTA

mers exploiting it. Most distributions set this up by default. In fact, you would be more likely to run into difficulty, if you wanted to allow the relay to forward internally generated mail. If you have Sendmail, you can edit the `/etc/mail/relay-domains` file; Qmail requires you to set the `RELAYCLIENT` environment variable, and Postfix uses the `mynetworks` variable in `main.cf`.

Receiving Mail

It is not a good idea to allow the individual host systems to receive incoming messages. You would need a domain for the email address of each device. Also, this would entail installing spam filters and virus scanners on every single machine. And let's not forget the slight, but undeniable, security risk that having port 25 (SMTP) listening on each system would pose.

The typical approach is similar to that adopted for outgoing mail. Incoming messages are bundled and handled by a small group of specialized machines. How that happens will depend to a great extent on your structure and the volume of mail you need to handle.

Control Center

One important question is, will you be using a central mail repository, or distributing mail to the individual machines? The centralized approach is simpler, and easier to secure. Distributing mail makes more sense in

environments with a high standard of technical qualification, as it allows the individual user more freedom of choice with respect to individual configurations. Also, users can harness the processing power of their workstations to apply additional filters and sorting rules.

In both cases, a central MX (Mail Exchange) host will handle incoming mail first. It can then forward messages for professional users directly to their workstations. You can disable the SMTP daemon, or restrict the daemon to receiving only local transmissions, for all other machines. The Post Office Protocol, POP, or the Internet Message Access

Protocol, IMAP, or in some cases a networking filesystem such as NFS, would then take care of distributing mail from the central mailbox system to the individual workstations.

Synchronized

Our previous examples all assumed a single relay and a single MX host for mail handling. In many small to midrange companies, both functions are typically performed by a single device. Experience tells us that a single computer can handle approximately 25,000 messages per day, or even more depending on local conditions.

Mail dispatching is typically more critical than mail reception. Although it does not need as much computing power, outgoing connections are often slower or even fail. This means that resources are tied up for longer. Also, MTAs like Qmail have trouble handling large mail queues. However, the ever-present flood of spam does mean that outgoing messages represent only a fraction of the total load. Ironically, this actually facilitates scaling.

Speeding systems

In the case of email reception, any filter mechanisms will impact system performance. The more filters you apply, and the more complicated these filters are, the longer it will take to receive a single message. This means that less messages can be handled simultaneously by a single computer.

Listing 1: Smart Host with Sendmail

```
define(`SMART_HOST', `relay.myorg.uk')dnl
FEATURE(`no_default_msa', `dnl')dnl
DAEMON_OPTIONS(`Port=smtp,Addr=127.0.0.1, Name=MTA')
FEATURE(`nocanonify')dnl
```

Listing 2: MX Configuration at the University of Trier

```
01 mas@ishi:~> dig uni-trier.de mx
02
03 ; <<>> DiG 8.3 <<>> uni-trier.de mx
04 [...]
05
06 ;; ANSWER SECTION:
07 uni-trier.de.          1D IN MX    10 rzmail.uni-trier.de.
08 uni-trier.de.          1D IN MX    50 rzmail2.uni-trier.de.
09
10 ;; AUTHORITY SECTION:
11 [...]
```

This makes selecting and programming filters extremely tricky. Procmail is useful for a single users who want to quickly and simply create a filter that will apply all kinds of products like Spam Assassin or anti-virus software. As soon as the volume reaches a critical level, however, Procmail turns into a killer: it spawns giant processes liberally, and really makes a meal of delegating its task. Sendmail, with its elegant and lean Milter mechanism [4] is preferable.

Distributed Load

If your mail server is overloaded, the first things to look for are inappropriate settings, such as in Procmail. But this is not always the answer. If you are having trouble with your MX host for incoming messages, it is quite simple to delegate additional MX hosts. Listing 2 shows the typical layered MX configuration, as used by the University of Trier. Note that a priority has been assigned to each MX server (to the left of the host name).

Mail dispatchers will attempt to use the machine with the lowest priority; thus *rzmail.uni-trier.de* (priority 10) can expect the most messages. The backup system, *rzmail2* (priority 50) will not be used unless the primary system is unavailable. Although this method pro-

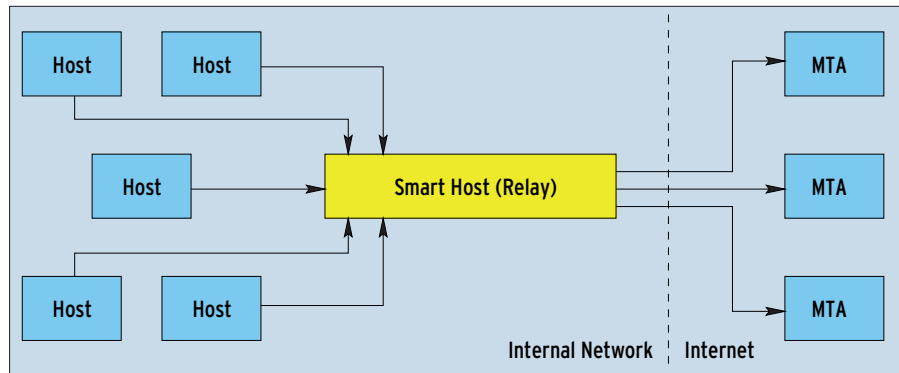


Figure 3: A central email relay or smart host takes care of forwarding outgoing messages

vides redundancy, it does not distribute the load. Each client and each MTA will attempt to send mail to *rzmail.uni-trier.de*.

AOL.com (see Listing 3) does this quite differently, combining two methods of load distribution. For one thing, the four MX hosts all have the same priority (15), which means that they will be used alternately by external sources. By entering the *dig* command multiple times with an interval between each, you see that the output order has changed.

Also, each of the four MX entries points to multiple IP addresses. This allows AOL to distribute incoming messages across a large number of machines to provide load balancing.

Reunited

AOL needs to pool the messages received by the individual servers before assigning them to customer mailboxes. The typical solution to this challenge is a fast fileserver that serves each of the mailboxes. Each of the MX hosts mounts this fileserver's mail directory and then writes any incoming messages directly to the mailbox of the appropriate user. Unfortunately, this approach is non-trivial: situations where two MX hosts attempt to deliver messages to the same user at the same time, and trip each other up in doing so, need to be avoided. We will be looking at this issue, and the possible solutions, in another article in this column.

The architecture we have been looking at also saves resources in other places. Whereas a single, central MX host needs to run daemons for POP or IMAP itself, it can now offload these to the fileserver or other computers. The second trick (allowing a single host name in DNS to point to multiple IP addresses) also helps to provide load balancing for your relay server. If this server is overloaded, simply add a second server with the same name. ■

Listing 3: MX-Configuration for AOL.

```
01 mas@ishi:~> dig aol.com mx
02
03 ; <<> DiG 8.3 <<> aol.com mx
04 [...]
05
06 ;; ANSWER SECTION:
07 aol.com.          1H IN MX   15 mailin-01.mx.aol.com.
08 aol.com.          1H IN MX   15 mailin-02.mx.aol.com.
09 aol.com.          1H IN MX   15 mailin-03.mx.aol.com.
10 aol.com.          1H IN MX   15 mailin-04.mx.aol.com.
11
12 ;; AUTHORITY SECTION:
13 [...]
14
15 ;; ADDITIONAL SECTION:
16 mailin-01.mx.aol.com. 5M IN A    64.12.138.152
17 mailin-01.mx.aol.com. 5M IN A    152.163.224.26
18 mailin-01.mx.aol.com. 5M IN A    205.188.156.122
19 mailin-01.mx.aol.com. 5M IN A    64.12.136.57
20 mailin-01.mx.aol.com. 5M IN A    64.12.137.89
21 mailin-01.mx.aol.com. 5M IN A    64.12.137.184
22 mailin-01.mx.aol.com. 5M IN A    64.12.138.57
23 [...]
```

INFO

- [1] Sendmail: <http://www.sendmail.org/>
- [2] Postfix: <http://www.postfix.org/>
- [3] Qmail: <http://www.qmail.org/>
- [4] Milter: http://www.sendmail.com/partner/resources/development/milter_api/;
Milter Perl modules:
<http://search.cpan.org/~cyng/>
- [5] Relevant RFCs: 821 and 2821 (SMTP), 1939 (POP3), 3501 (IMAP)
<http://www.rfc-editor.org>