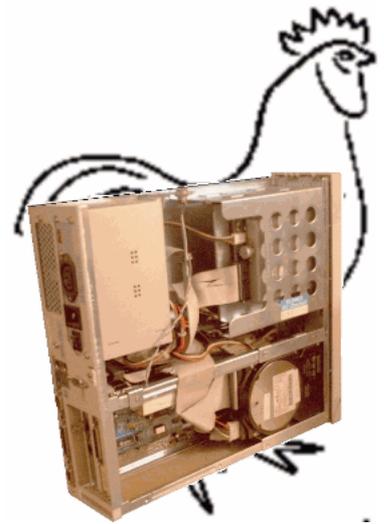


## Salvage Server Project Report 3:

# Setting Up Mail Forwarding Lists With KMail

Produced by the Free Range Salvage Server Project, October 2003  
 email: fraw@gn.apc.org web: <http://www.fraw.org.uk/ssp/>

**KMail is a moderately lightweight email client that works well under both the KDE and Gnome desktops. But it also has very good mail filtering functions. These can be manipulated to forward mail automatically each time your mail is checked. So making KMail function as a small email list server.**



This idea started out as solution to a problem – how to create short-term email lists. There are free list servers available, like Yahoo's *egroups*. But you have to accept advertising appended to every mail sent through the list. You could buy a list on a server, but that costs money, and often it's an excessive answer when you just want to set up a short-term list to support an event.

Linux has many facilities to set up mailing lists on your own computer – although this can be complex if you are not your own ISP. So we need something far simpler that anyone with a little experience of using email can set up.

The simplest option is to use the features of an existing email program. Whilst there are many options available, we began experimenting with the mail filtering functions within KMail to set up simple mail forwarding. The advantage of this is that you don't have to set up a server. Each time that you check your email the program will identify mail that is to be forwarded, and then automatically send it out as your other ordinary mail comes down.

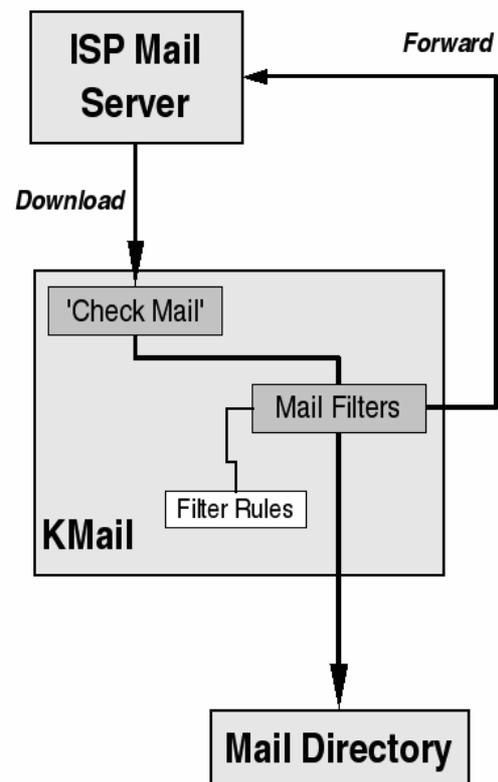
This is not a solution that is acceptable as a replacement for an email list server. There are two significant flaws. Firstly, it only functions when you check your mail – forget to check your mail and the messages will not be relayed. Secondly, if the mail ever bounces the bounces could propagate repeatedly over the list until you notice the problem and manually remove that address from the list. Thirdly, there's no facility to archive messages from the list. But, on the positive side, as a means of setting up a short term list around a particular event or project, it's far more flexible than using an email list, or free list service.

## Options for identifying lists – by address or subject

For this system to work you have to have an email address to send mail for forwarding to. This needs to be different from your ordinary email address. If you use the same

address for everything then mail will just 'feedback' through the mailbox. You must download mail from the list address, and forward under the list address. But for you to participate in the list you have to send under a different address to that of the list.

### Forwarding With KMail



There are two options:

- ◆ Have two email addresses (quite common today with most ISPs offering dial-up email) – which allows multiple forwarding lists if the mail was identified with a different

subject line. If you want more than one list with this option list users must correctly type the list name in the subject line – e.g. '[LIST]'.

- ◆ For those with a domain name that forwards mail for the domain to their usual email address, use can any address you like within your domain – you then filter the email from your ordinary mail.

Of the two, the latter is the best option. This is because each forwarding list can have a different – albeit non-existent – email address. Your ISP usually forwards email from the domain to your main email address – providing the required separation between your email address and the list address.

Identifying different lists by the subject line requires that people remember to correctly type the name into the subject line. If this isn't done, the mail won't forward. You could do without a subject line ID, but that will limit you to operating only one forwarding list because you can't differentiate between them.

### KMail user IDs

To operate forwarding using KMail you need to set up a user ID for the list and a mail filter. The purpose of the user ID is to provide an identity for the email list when KMail forwards mail. If you don't do this then the mail will be

forwarded under your own identity – meaning that any replies will only go back to you.

From the KMail menu bar, click *Settings*, then *Configure KMail*, and if it's not already there, click on the *Identities* tab. To create a new mail ID click *New*. In the *New Identity* box, type the name of the forwarding list (see 'a' below).

Clicking 'OK' takes you to the ID configuration dialog box at the *General* tab (see 'b' below). Firstly, enter the name of the list in the *name* box, and the email address in the *address* box. Next to the *advanced* tab (see 'c' below). Set the *reply-to address*, which should be the same as the email address. If you wish you can set the locations where mail will be stored, but this is not necessary as these will be set via filtering later. You need not set anything in the *signature* tab (see 'd' below), so just click OK to set the new identity.

This completes the configuration of the identity. If you use email addresses from a domain you need to set up one ID for each address. If using a single address with a subject line ID, you need only set up one ID.

### KMail's mail filters

Mail filters are a time saving means of sorting your email according to who they're from or dumping obvious spam

straight into the trash. But KMail's filters have powerful features that allow us to re-write the headers of the email and send it out again.

From the KMail menu bar, click *Settings*, then *Configure Filters*. To create a new filter click the *New* button at the bottom left of the dialog box. We'll begin by setting up a domain-based forwarding list.

To begin, click on the *More* button in the *Filter Options* part of the dialog box. You need seven filter options – click *more* until you have seven. Then, in the *Advanced Options* area, uncheck the *on manual filtering* and if



(a)



(c)



(b)



(d)

*filtering matches* options – just leave the *to incoming* option checked. This ensures that the filter will only work on downloaded messages.

Firstly we set the *Filter Criteria*. Make sure that *Match all* is selected. To filter on the incoming email address we set the first selection box on the first line to *To*. This tests every *To:* header on each incoming email to see where it was sent to. The next selection box should be set to *contains*. In the text box we enter the email address of the email forwarding list – in the example above, *newlist@domain.org.uk*. This name must correspond to the email address given in the KMail ID you set up previously for the list.

Next we set the *Filter Actions*. Filtering is a chain of events triggered when the criteria in *Filter Criteria* are met. This could be one action – such as moving the email to a particular directory. But for email forwarding we have seven actions – one action per line:

1. We remove the *Reply-to* header from the email. Click on the first selection box and select *remove header*. The next selection box on the line will now change its format. Click on this and select *Reply-to*. This will be set to the email address of the email sender. We remove it to replace it with the email address of the list.
2. We change the identity of the outgoing email to that of the list. This ensures that the *Reply-to* header on the outgoing email belongs to the list, not the ordinary KMail client user. Click the first selection box and select *set identity to*. The next box will now contain a list of configured KMail users, and from this box you select the ID you configured for the forwarding list.
3. Next we edit the return path to make sure that it doesn't contain the user's details, but those of the email list. Set

the first selection box to *rewrite header*. The second selection box doesn't contain an entry for *Return-path* and so we manually type this into the box. Then enter your email address in the first text box, and the email address of the list in the second.

4. We rewrite the *From:* header to ensure that it contains the identity of the list, changing the user's email address to that of the list. Again, *rewrite header* doesn't have an entry for *From*, so type it into the blank option. Then enter your usual email address in the first text box and the list email address in the second.
5. To complete the rewrite of *From:*, we change the name from the user's to that of the list – follow the same procedure as above, but enter the name KMail uses for your ordinary account in the first box, and the name of the list in the second.
6. Now we redirect the email to a list of users. In the first selection box select *redirect to*. You are then given a text box. It is this text box that will contain the list of people that the email on the list will be sent to. Whenever you need to change list member details, you have to edit the contents of this box. To check that the forwarding filter works, you need to include your own address here.

7. If we do nothing more the email that has been forwarded will be left in your *inbox* folder. We conclude by moving to the incoming email to a folder created to hold the email that has been forwarded to the list.

Clicking *OK* stores the filter rule. You can now test if it works by emailing to the list address. Then check your mail. You should notice that during the mail check the counter for the *outbox* folder clocks up a pending mail, and then goes back down again once its sent.

Check your mail again to see if the forwarding filter works. Select *View* from the menu bar, then *Headers*, then *All Headers*. Now view the email and check the content of the headers. If all's well the identities in the headers should relate to the list ID, not your main email ID. The return addresses should also point to the list email address.

If you wish to use a single address, and then filter on an ID in the subject line, you need to configure the filter rule differently (see box above). In addition to the *Filter Criteria* for the email address, you need to add a second that checks the subject line for the list ID. Other than this, the *Filter Actions* section carries out the same changes to the content of the mail. But instead of having a different address for each list, you use the same email address and have a different filter rule for each of the forwarding lists that you run through it. The only difference between each being the name in the *Filter Criteria* test of the *Subject* line, and the content of the *redirect to* box.

You will have to completely write the filter rule from scratch each time you create a new list as there's no copy function for filter rules.

### Beware – this is a cheap and easy hack

This idea has been developed to solve a simple problem – getting a small automated list to co-ordinate people on-line up and running without having to resort to a full-blown list server. But it's not without its problems.

As well as the fact it only works when you check your mail, it is particularly susceptible to bounced mail. But the fact that you manually trigger the forwarding process when you check mail means that, in between checking mail, if you check the folder where you move forwarded mails you should easily see problems when they develop.

The other restriction is that the list is wholly public. Unless you add additional *Filter Criteria* to test the *From* header in each mail, and only trigger for certain email addresses or domains, then anyone can send through the list once its address is known. As we can only redirect within the *To*: header, not the *Bcc*: header, it also means that the distribution list will be mailed each time in the *To*: header. This makes it a potential spam liability, and potentially a conduit for viruses when Windows users are part of the list.

The Salvage Server Project has been developed by the Free Range Network to promote the use of redundant IT equipment as a resource for community and grass roots campaigning organisations. This report has been produced to support the work of the project, and is made freely available to encourage the objectives of the project.

© Copyright 2003, Paul Mobbs/Free Range Network. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with Invariant Sections being the document title and author identification, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is provided at: [http://www.fraw.org.uk/\\_admin/rights.html](http://www.fraw.org.uk/_admin/rights.html) This document has been wholly produced using the Gnu/Linux operating system and free software.