

# MAC OS X HACKS

*100 Industrial-Strength Tips & Tools*



O'REILLY®

*Rael Dornfest & Kevin Hemenway*

# MAC OS X HACKS

# MAC OS X HACKS



*Rael Dornfest and Kevin Hemenway*

**O'REILLY®**

Beijing • Cambridge • Farnham • Köln • Paris • Sebastopol • Taipei • Tokyo

HACK  
#82

## Getting sendmail Up and Running

sendmail is powerful, but at times it appears complicated too. Unravel the sendmail knot and you can configure this awesome mail server on your Mac OS X system.

sendmail is complicated software, no doubt about it. But sendmail is also the Swiss Army Knife of mail servers, and I don't mean one of those little keychain trinkets. Instead, it's the monster three-inch-wide kind with all the tools, most of which you have never seen before and have no idea what they do. However, with a little time and patience, you too can become proficient enough with sendmail to make it accomplish everything you need.

Here's what I'll cover in this hack:

- Dealing with Jaguar's permissions and sendmail's security precautions
- Working with configuration files
- The `USER_RELAY`
- How to set up aliases
- How to allow relaying from certain hosts
- Running behind a firewall
- Working with lame ISPs

Be warned, this is not a beginner's article. If you're uncomfortable performing shell commands as root on your system with `sudo` [Hack #50] and editing special Unix files [Hack #51], you may want to acquaint yourself with these first. However, if you do have a bit of shell experience, and I haven't scared you off by mentioning the word `pico`, then this should be just the quick reference you need to get your own mail server running under OS X.

### Hacking the `hostconfig` File

First, we'll need to edit the `MAILSERVER` line in `/etc/hostconfig` so that sendmail starts automatically:

```
% sudo pico /etc/hostconfig
```

The file will load. Use the arrow keys to navigate the file and edit the `MAILSERVER` line to look like this:

```
MAILSERVER=-YES-
```

If we lived in a perfect world, our next step would be to start sendmail. However, sendmail is a somewhat tricky beast to work with.

## On Permissions and Blame

The number one trick to setting up sendmail on Mac OS X is dealing with the way that Apple has configured the permissions on the various directories of the filesystem. You see, in its quest to make Unix more Mac-like, Apple decided that it would be best to allow users, at least administrative users, to be able to move files in and out of the root directory with impunity. Apparently Apple doesn't want users to see a "You can't drag that file here!" dialog box.

This clashes heavily with sendmail's built-in paranoia. You see, sendmail really wants any directory that it is involved with to be modifiable only by the root user. This includes the `/` and `/Users` directories. It will complain bitterly and refuse to start up with a statement that looks something like this:

```
/etc/mail/sendmail.cf: line 93: fileclass: cannot open '/etc/mail/local-host-names': Group writable directory
```

There are two primary solutions to this problem:

- Change the ownership of the `/` and `/Users` directories to something that sendmail prefers (`chmod g-w / /Users`).
- Configure sendmail to ignore its instincts and operate even though the permissions on some folders aren't exactly as it likes.

The first of these solutions is a bit more extreme, but it is the safest way to set up your server. It is the correct solution for the paranoid system administrator who wants to make sure that nobody, not even any of her users, can compromise the system. It does have the side effect that nobody, not even the administrator, will be able to use the Finder to copy files into the `/` and `/Users` directories.

On the other hand, as long as you trust every person you give a user account to (or at least every user that you allow to administer your machine), there is a better way to go about this. This is to use the `DontBlameSendmail` configuration parameter with sendmail. Think of it as administering a small amount of medication to sendmail to reassure it that not everything in the world is a risk. For most people running Mac OS X (who aren't admins of systems serving hundreds or thousands of potential users and don't have untrusted or unknown users on the machine), this is the appropriate strategy to use.

In order to implement this solution, we're going to have to dig into how to work with sendmail's configuration files.

## Working with Configuration Files

As soon as you decide to work with sendmail's configuration files, you'll find out that there is a lot of confusing stuff in there.

## Getting sendmail Up and Running

Take a look at the */etc/mail/sendmail.cf* file. The first thing you see is a header that says:

```
##### DO NOT EDIT THIS FILE! Only edit the source .mc file.
```

Scroll down a bit further and you'll see some stuff that could look friendly only to an old-time Perl hacker:

```
# hostnames ending in class P are always canonical
R$* < @ $* $=P > $* $: $1 < @ $2 $3 . > $4
R$* < @ $* $~P > $* $: ${daemon_flags} $| $1 < @ $2 $3 > $4
R$* CC $* $| $* < @ $+.$+ < $* $: $3 < @ $4.$5 . > $6
R$* CC $* $| $* $: $3
# pass to name server to make hostname canonical
R$* $| $* < @ $* > $* $: $2 < @ $[ $3 $] > $4
R$* $| $* $: $2
```

So, if you're not supposed to edit this file, and really wouldn't want to even if you should, what are you supposed to do? The answer is to ignore it. Treat it like a binary file. You don't muck about in the */bin/sh* executable to use it do you? Take the same approach to */etc/mail/sendmail.cf*.

Instead, we're going to see how to edit the source code for this file. Take a look at the */usr/share/sendmail/conf/cf/generic-darwin.mc* file. The body of it looks like this:

```
VERSIONID(`Id: generic-darwin.mc,v 1.3 2002/04/12 18:41:47 bbraun Exp $')
OSTYPE(darwin)dnl
DOMAIN(generic)dnl
undefine(`ALIAS_FILE')
define(`PROCMAIL_MAILER_PATH',`/usr/bin/procmail')
FEATURE(`smrsh',`/usr/libexec/smrsh')
FEATURE(local_procmail)
FEATURE(`virtusertable',`hash -o /etc/mail/virtusertable')dnl
FEATURE(`genericstable',`hash -o /etc/mail/genericstable')dnl
FEATURE(`mailertable',`hash -o /etc/mail/mailertable')dnl
FEATURE(`access_db')dnl
MAILER(smtp)
MAILER(procmail)
```

This is much more approachable than *sendmail.cf* ever could be. This is actually a script written in the m4 macro language. m4 has been around for a while and Mac OS X ships with GNU m4 Version 1.4. Luckily, it is simple enough to use without having to learn much about it. If you are interested in learning more, see the GNU m4 project page (<http://www.gnu.org/software/m4/m4.html>).

So, this is the source code we'll use to configure sendmail. Let's make a copy of it and put it where we will remember where it is:

```
% sudo cp /usr/share/sendmail/conf/cf/generic-darwin.mc /etc/mail/config.mc
```

We now have a copy of the source code for the `sendmail.cf` file in a place where we can edit it and keep track of its location. However, even if you have a copy of the source code, you still have to know how to compile the file. In our case, the set of commands to compile the `config.mc` file to `sendmail.cf` are:

```
% m4 /usr/share/sendmail/conf/m4/cf.m4 /etc/mail/config.mc ␣
> /tmp/sendmail.cf
% mv /etc/mail/sendmail.cf /etc/mail/sendmail.cf.old
% mv /tmp/sendmail.cf /etc/mail/sendmail.cf
```

Yikes! That's too much to remember. It goes against my philosophy of keeping things as simple as possible (without being too simple, that is!). Luckily, I've written a little script that should help this part of working with sendmail's configuration files.

## A Script to Simplify Your Life

The following script will compile `config.mc` into `sendmail.cf` and restart sendmail so that it will notice the configuration changes. You can put it anywhere you want; I happen to have placed my copy in the `/etc/mail` folder so that I can find it easily. Fire up your editor of choice and type this in. Then, if you want to mirror what I've done, save it to `/etc/mail/update`. Otherwise, you may want to save it as `/usr/local/bin/sendmail-update` or some other fairly easy-to-remember location.

```
#!/bin/sh
if [ /etc/mail/config.mc -nt /etc/mail/sendmail.cf ]
then
echo Regenerating sendmail.cf
m4 /usr/share/sendmail/conf/m4/cf.m4 /etc/mail/config.mc > \
/tmp/sendmail.cf
mv /etc/mail/sendmail.cf /etc/mail/sendmail.cf.old
mv /tmp/sendmail.cf /etc/mail/sendmail.cf
/System/Library/StartupItems/Sendmail/Sendmail restart
fi
```

We're going to add a bit more to this script later. But for now, we're ready to feed sendmail that antiparanoia medicine.

## Back to DontBlameSendmail

In order to use the `DontBlameSendmail` configuration parameter with sendmail, all we need to do is add one line to the `config.mc` file.

Edit it to match the following code. The line you need to add is boldfaced.

```
% sudo emacs /etc/mail/config.mc

VERSIONID(`$Id: generic-darwin.mc,v 1.3 2002/04/12 18:41:47 bbraun Exp $')
```

## Getting sendmail Up and Running

```

OSTYPE(darwin)dnl
DOMAIN(generic)dnl
undefine(`ALIAS_FILE')
define(`PROCMAIL_MAILER_PATH',`/usr/bin/procmail')
define(`confDONT_BLAME_SENDMAIL',`GroupWritableDirPathSafe')
FEATURE(`smrsh',`/usr/libexec/smrsh')
FEATURE(local_procmail)
FEATURE(`virtusertable',`hash -o /etc/mail/virtusertable')dnl
FEATURE(`genericstable',`hash -o /etc/mail/genericstable')dnl
FEATURE(`mailertable',`hash -o /etc/mail/mailertable')dnl
FEATURE(`access_db')dnl
MAILER(smtp)
MAILER(procmail)

```

Be careful to note that the quoting uses both the backtick (`) and single quote (') characters around the arguments to the define statement. Save the file. Next, we need to compile it. Execute your update script. You may need to remember to give it execute permissions (`chmod g+x /etc/mail/update`) first!

```
% sudo /etc/mail/update
```

```

Regenerating sendmail.cf
Restarting mail services

```

After sendmail restarts, you can verify that it is running properly by trying the following in a Terminal window:

```

% telnet localhost 25
Trying 127.0.0.1...
Connected to dsl092-007-021.sfo1.dsl.speakeasy.net.
Escape character is '^]'.
220 dsl092-007-021.sfo1.dsl.speakeasy.net ESMTP Sendmail 8.12.2/8.12.2; Sat,
10 Aug 2002 00:43:35 -0700 (PDT)
QUIT
221 2.0.0 dsl092-007-021.sfo1.dsl.speakeasy.net closing connection
Connection closed by foreign host.

```

Simply type QUIT to end the interactive session. sendmail is now up and running. It will accept mail addressed to any user at the local host. For example, on my server, sendmail will accept any mail addressed to *duncan@dsl092-007-021.sfo1.dsl.speakeasy.net* (my ISP's address), but not *duncan@somehost.dyndns.org* (my personal web site I want to accept mail from). This is a good start and shows that the mail server isn't an open relay that will possibly spread spam, but we need to do a little more configuration to allow us to accept mail to our desired hostname.

## Telling sendmail Which Hostnames Are Valid

To have sendmail accept mail sent to your machine's hostname, all you need to do is edit the */etc/mail/local-host-names* file. To do so, enter the following command:



```
% sudo pico /etc/mail/local-host-names
```

Simply add the hostnames that you want to receive mail for, one line at a time, to this file. For example:

```
somemachine.dyndns.org
66.92.7.21
```

For this to take effect, you'll need to restart sendmail. Instead of rebooting, we're simply going to stop and restart sendmail. Use the following command to do so:

```
% /System/Library/StartupItems/Sendmail/Sendmail restart
```

## Setting Up the LUSER\_RELAY

The next setting we are going to look at is the LUSER\_RELAY. No, this doesn't mean a way to deal with those 14-year-old kids who hold their hands up to their foreheads saying "loooooos-errrr," but instead is a way of handling email that comes to your server that is not addressed to any user. The LUSER\_RELAY setting will direct any piece of mail to your server without a user to a particular user's account.

This is particularly handy when you want to be able to hand out lots of different addresses, such as *im-a-geek@myhost.com* and *spam-target@myhost.com*, without having to set up anything on your server. I personally use this feature all the time when giving my email address out to stores that I'm interested in getting email from but fear that they will sell the address off or pummel me with too much information later.

So, to set this up, simply edit the *config.mc* file as follows (the bold line is where you will replace *duncan* with the name of the local user you want to get the mail!):

```
% sudo pico /etc/mail/config.mc
```

```
VERSIONID(`$Id: generic-darwin.mc,v 1.3 2002/04/12 18:41:47 bbraun Exp $')
OSTYPE(darwin)dnl
DOMAIN(generic)dnl
undefine(`ALIAS_FILE')
define(`PROCMAIL_MAILER_PATH',`/usr/bin/procmail')
define(`confDONT_BLAME_SENDMAIL',`GroupWritableDirPathSafe')
define(`LUSER_RELAY',`local:duncan')
FEATURE(`smrsh',`/usr/libexec/smrsh')
FEATURE(local_procmail)
FEATURE(`virtusertable',`hash -o /etc/mail/virtusertable')dnl
FEATURE(`genericstable',`hash -o /etc/mail/genericstable')dnl
FEATURE(`mailertable',`hash -o /etc/mail/mailertable')dnl
FEATURE(`access_db')dnl
MAILER(smtplib)
MAILER(procmail)
```

## Getting sendmail Up and Running

Now, just run the update script:

```
% sudo /etc/mail/update
Regenerating sendmail.cf
Restarting mail services
```

Try things out. Use your mail client to send mail to all sorts of addresses that don't exist on your machine.



This all assumes you've already set yourself up with [Domain Name Service \[Hack #78\]](#) and have an `mx` (mail) record pointing at the Mac.

## Allowing Relaying from Certain Hosts

sendmail doesn't like to relay mail that isn't sent from trusted sources. The designers of sendmail do this purposefully to try to alleviate the problem of spam. You see, spammers take advantage of mail servers that will relay mail from anyone in order to send mail to all of us while taking advantage of somebody else's bandwidth costs. It's truly heinous.

By default, sendmail's paranoia means that when we set up a server, we can relay through it only mail that originates on the local machine. In order to use it as a proper mail server, we need to let it know which hosts to trust to relay mail. For example, my mail sever is configured to accept email that comes from my private home network that is running behind a Network Address Translation (NAT) with a fixed IP address. In addition, I always want to be able to send mail, using my laptop, from my friends' houses that have known hostnames. To do this, you simply need to define these rules in the `/etc/mail/access` file, as shown here:

```
% sudo pico /etc/mail/access

192.168.123.2 RELAY
dsl-1-1-1-1.networkprovider.net RELAY
```

You can also allow blocks of IP addresses or partial domain addresses to relay through your server. For example, to allow anybody on a subnet, as well as let everybody at the *oreilly.com* domain use my mail server, I could edit this file to look like this:

```
% sudo pico /etc/mail/access

192.168.123.2 RELAY
dsl-1-1-1-1.networkprovider.net RELAY
192.168.145 RELAY
oreilly.com RELAY
```

This will let anyone with an IP address that starts with 192.168.145 or whose IP address resolves to the *oreilly.com* domain to use our server. We need to compile this file into a form that sendmail can use. To do this, use the following command:

```
% sudo makemap hash /etc/mail/access > /etc/mail/access
```

Yes, this is yet another command to remember, and I personally always have to look it up to use it. Don't fear; we can fix this problem.

## Our Helper Script Expanded

Since I hate having to use the documentation to execute what should be simple commands, I have actually added all these commands (and more) to my update script. I gave you the short form earlier. Here's the long form (with the section we haven't seen before in bold type):

```
#!/bin/sh

if [ /etc/mail/config.mc -nt /etc/mail/sendmail.cf ]
then
echo Regenerating sendmail.cf
m4 /usr/share/sendmail/conf/m4/cf.m4 /etc/mail/config.mc > \
/tmp/sendmail.cf
mv /etc/mail/sendmail.cf /etc/mail/sendmail.cf.old
mv /tmp/sendmail.cf /etc/mail/sendmail.cf
/System/Library/StartupItems/Sendmail/Sendmail restart
fi

if [ /etc/mail/access -nt /etc/mail/access.db ]
then
echo Updating access
makemap hash /etc/mail/access < /etc/mail/access
fi
```

In short, this file checks to see if it should:

- Compile the *sendmail.cf* file
- Update the access database

When the source for any of these files is out-of-date, it will be updated. Easy huh? Now, all we have to do is remember to run the update script whenever we edit one of the configuration files and the right thing will happen.

## Running Behind a Firewall

Running sendmail behind a firewall, especially if it's a NAT, can confuse it. You see, sendmail does its best to try and figure out what its hostname is. As long as your machine is a first-class citizen on the Internet (i.e., it has an IP address visible from the Internet at large), it can usually do a good job at

## Getting sendmail Up and Running

this. However, when you are running behind a NAT or if your IP address doesn't resolve to any hostname, you'll need to give sendmail a little help. For example, if you are hosting mail for *domain.com*, you need to tell sendmail that its domain name is *\$w.domain.com*. The *\$w* part is an important part of sendmail trickery that means insert the local hostname here.

To configure sendmail to use a specific domain name, edit your */etc/mail/config.mc* file as follows:

```
% sudo pico /etc/mail/config.mc

VERSIONID(`$Id: generic-darwin.mc,v 1.3 2002/04/12 18:41:47 bbraun Exp $')
OSTYPE(darwin)dnl
DOMAIN(generic)dnl
undefine(`ALIAS_FILE')
define(`PROCMAIL_MAILER_PATH',`/usr/bin/procmail')
define(`confDONT_BLAME_SENDMAIL',`GroupWritableDirPathSafe')
define(`LUSER_RELAY',`local:duncan')
define(`confDOMAIN_NAME',`$w.domain.com')
FEATURE(`smrsh',`/usr/libexec/smrsh')
FEATURE(local_procmail)
FEATURE(`virtusertable',`hash -o /etc/mail/virtusertable')dnl
FEATURE(`genericstable',`hash -o /etc/mail/genericstable')dnl
FEATURE(`mailertable',`hash -o /etc/mail/mailertable')dnl
FEATURE(`access_db')dnl
MAILER(smtp)
MAILER(procmail)
```

As always, remember to run the update script:

```
% sudo /etc/mail/update
Regenerating sendmail.cf
Restarting mail services
```

Next, we'll take a look at one other common problem that people have that is introduced by their ISP.

## Working with Lame ISPs

What do I mean by lame ISPs? Well, I mean ISPs that block all outgoing traffic on port 25. Instead of letting you have access to the Internet on port 25, they want you to use only their own mail server. They do this to try to stop spammers from utilizing open relays on their networks. However, this means that your personal mail server can't send mail to other hosts on the Internet.

Luckily, since sendmail is the Swiss Army Knife of mail servers, there is a configuration directive to fix this. To have all mail from your server go through your ISP's mail server, edit your */etc/mail/config.mc* file to match the following code:

```
% sudo pico /etc/mail/config.mc
```

```

VERSIONID(`$Id: generic-darwin.mc,v 1.3 2002/04/12 18:41:47 bbrown Exp $')
OSTYPE(darwin)dnl
DOMAIN(generic)dnl
undefine(`ALIAS_FILE')
define(`PROCMAIL_MAILER_PATH',`/usr/bin/procmail')
define(`confDONT_BLAME_SENDMAIL',`GroupWritableDirPathSafe')
define(`LUSER_RELAY',`local:duncan')
define(`confDOMAIN_NAME',`$w.domain.com')
define(`SMART_HOST'`mail.mindspring.com')
FEATURE(`smrsh',`/usr/libexec/smrsh')
FEATURE(local_procmail)
FEATURE(`virtusertable',`hash -o /etc/mail/virtusertable')dnl
FEATURE(`genericstable',`hash -o /etc/mail/genericstable')dnl
FEATURE(`mailertable',`hash -o /etc/mail/mailertable')dnl
FEATURE(`access_db')dnl
MAILER(smtp)
MAILER(procmail)

```

Once again, run the update script:

```

% sudo /etc/mail/update
Regenerating sendmail.cf
Restarting mail services

```

Problem solved.

## Getting NetInfo Out of the Picture

Some of the Apple documentation on sendmail (notably, the */etc/mail/README* file) implies that it's a good idea to set a few properties in NetInfo to ensure that the sendmail binary reads its configuration from */etc/mail/sendmail.cf*. So far, I've not had a problem with this, but in the interest of making sure that we don't get bit by a modified sendmail binary from Apple in the future, we should go ahead and execute the commands that will modify the NetInfo database:

```

% sudo niutil -create . /locations/sendmail
% sudo niutil -createprop . /locations/sendmail sendmail.cf ↵
/etc/mail/sendmail.cf

```

—James Duncan Davidson