

The Sysadmin's Daily Grind: *imapsync*

# Mail Mover

If you need to set up a new mail server, use the handy *imapsync* tool to migrate the contents of any existing IMAP mailboxes. *imapsync* synchronizes the new and old mail servers, preserving mailbox settings so users won't notice the change. **BY CHARLY KÜHNAST**

In a perfect world, users do not notice a mail migration. Unfortunately, perfection is rarely the case in production environments. One problem is that migration routines tend to reset read/unread flags, leaving the user with an inbox where everything is tagged as "new." The Perl script *imapsync* [1] solves this problem by synchronizing IMAP accounts between servers.

## Practicalities

I have a tired old mail server with an IP address of 10.0.0.5 and a new mail server with the address 10.0.0.10. My username on the old machine is *charly*, and my username on the new machine is *ckuehnast*. I will begin by simulating a mail migration using *imapsync*'s *--dry--* option:

```
imapsync --dry --host1 10.0.0.5
--user1 charly --password1
Orland0 -- host2 10.0.0.10
--user2 ckuehnast --password2
10nd0n
```

The *--dry* option tells *imapsync* to execute all the steps without actually

committing anything to disk. Assuming I am happy with the results of the simulation, I can run the command without the *--dry* option to perform the actual migration.

The preceding command creates a security risk by directly passing my passwords (Orland0 and 10nd0n) to *imapsync*. A user logged on to the machine could type *ps auxwww* and grab the passwords off the process list. A better solution is to store my passwords in a file with restrictive permissions (*chmod 400 filename*). I could then use the following command:

```
imapsync --dry --host1 10.0.0.5
--user1 charly --passfile1
/etc/charly-pass --host2
10.0.0.10 --user2 ckuehnast
--passfile2 /etc/ckuehnast-pass
```

Unfortunately, *imapsync* must still transmit the password as clear text in order to log on to the target server. *imapsync* does not support IMAP via SSL, and this puts admins at a disadvantage against sniffers.

## Clever Boy

*imapsync* does not simply transfer the whole IMAP account. Instead, it inspects the target server to see if any mail is in the account. If so, *imapsync* performs a differential migration to synchronize the accounts. If an interruption occurs during the transfer, the tool simply restarts and picks up from where it left off.

If you are sure you will not need the messages on the old server in future, you can pass the *--delete* parameter to *imapsync*. *--delete* removes the messages from the source server after the migration.



Deutsche Post World Net

Incidentally, the author of *imapsync* also has another useful tool called *Pop2imap* [2]. *Pop2imap* migrates POP3 accounts to IMAP servers. According to the official description of the tool, "Pop2imap is a tool for incremental POP to IMAP transfer from one mailbox to another." Like *imapsync*, *Pop2imap* reduces the amount of data transferred across the network by only copying messages that are not present on both servers. If you are accustomed to running *imapsync*, you should have no trouble at all migrating your POP3 accounts using *Pop2imap*. ■

## INFO

[1] *imapsync*: <http://www.linux-france.org/prj/imapsync/>

[2] *Pop2imap*: <http://www.linux-france.org/prj/pop2imap/>

## SYSADMIN

### Admin Workshop .....58

A good place to start with troubleshooting is the system process list. This month we look at the processes running on a typical Linux workstation soon after startup.

### PHP Security .....62

PHP is safer than an ordinary CGI script, but safer may not be safe enough. We'll show you how to build security into your PHP-based Web pages.

## THE AUTHOR

Charly Kühnast is a Unix System Manager at the data-center in Moers, near Germany's famous River Rhine. His tasks include ensuring fire-wall security and availability and taking care of the DMZ (demilitarized zone).

