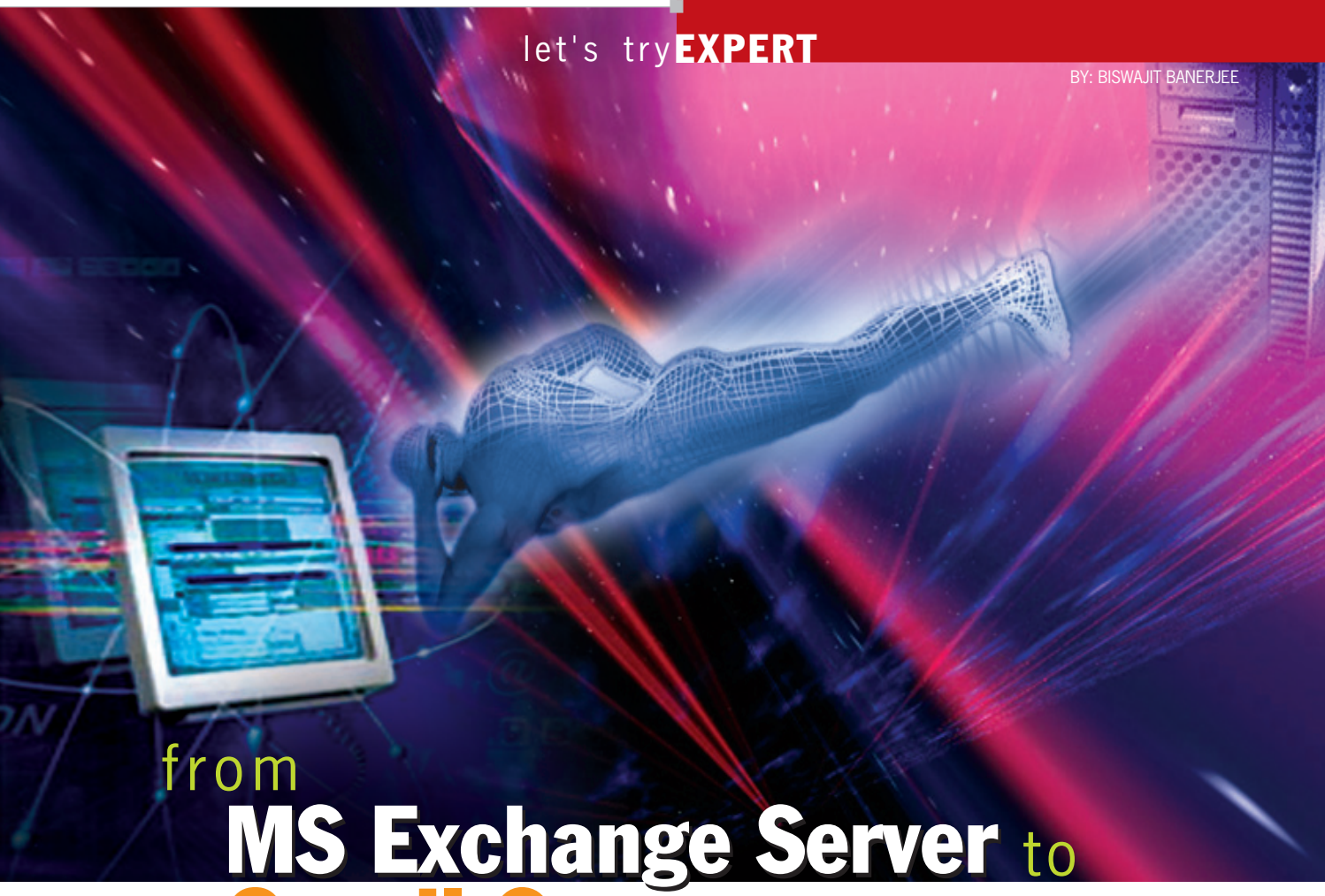
BY: BISWAJIT BANERJEE



from MS Exchange Server to Qmail Server

How can Microsoft Exchange Server migrate to the back-end Qmail server without inconveniencing present mail users?

Any e-mail server requires extensive flexibility to serve the needs of modern organisations today. In my previous article on 'Qmail Server in the Enterprise' carried in the March 2003 issue of LINUX For You I had discussed building up an e-mail server for the enterprise, based on Dan Brenstein's Qmail MTA. In this article, I will be discussing the migration from Microsoft Exchange Server to the back-end Qmail server and issues related to its implementation. For understanding the whole migration, we need to look into the working and the facilities provided by the Microsoft Exchange Server.

At this stage, I would like to point out the criticalities of migration. Any intended migration should take place with minimal changes in the user interface so that users may find it easy to work on the new one. Organisations would have already spent a lot of money, time and training during the course of deployment of their earlier mail application. Any additional changes, as a result of migration, can add to user dissatisfaction, training costs, and can also result in a fall in the user's productivity. We must also keep in mind that e-mail is a critical application for an organisation. Migration of e-mail application servers has to be planned carefully so that it causes minimum inconvenience to the user community.

THE NEED FOR OPEN SOURCE

- As the number of e-mail users grows, the cost factor becomes critical. If solutions are

available on open source, where licensing is not an issue, why shouldn't any organisation or corporate opt for it? For example, mantraonline.com, a Delhi- and Mumbai-based service provider, has migrated all its dial-up users to the Qmail + LDAP environment, since the Netscape e-mail box licences cost more than dial-up Internet access.

- In case of e-mail servers such as Microsoft Exchange, organisations have no option with regard to the functionality of the product—they have to accept what the vendor provides. But in case of open source solutions such as Qmail, the user can customise the software according to the changing e-mail requirements of the organisation. In other words, proprietary solutions limit flexibility—either you compromise with what is there or invest in some other product.

A typical example of this is the Web interface provided by Microsoft Exchange. Generally, organisations do not choose using this interface, as it is too slow. MS Exchange Server does not allow any other Web interface or Web client due to its proprietary nature, whereas open source solutions such as Qmail or Sendmail provide ample choice to organisations in terms of speed, features and flexibility.

- A fast, scalable and trouble-free system is essential for e-mail applications. Here, the operating system of the e-mail application server plays a vital role. No doubt Windows 2000 has a number of great features. But these lag behind the features of the Unix operating system. Open source solutions such as Qmail and Sendmail are optimised for Linux and Unix-type operating systems. The Unix operating system is built to handle high traffic and volumes.
- Larger organisations usually need clustering and load balancing. If there is an increase of traffic and load on the server, the performance of the e-mail application is bound to suffer. Concepts of clustering and load balancing play an important role in the choice of the e-mail server. Linux clustering and load balancing technology has established itself in the world of critical and real-time applications.
- Microsoft Exchange is an application that needs high-end resources to be deployed. As the number of users increases or the traffic rises, hardware resources need upgradation, which involves high expenditure and, sometimes, demand-forced migration. Upgradation of hardware may not be difficult, but migration always is. In case of a Qmail-based environment, there is a high degree of flexibility where you need to change the set-up for a long period. Further, Qmail is not resource-hungry.
- Since Microsoft Exchange is not available on other operation systems, it binds and restricts its portability. It does not support open source platforms such as LDAP. In the present scenario, open source solutions are more attractive, at least in terms of connectivity and distributed services.
- A user of the e-mail application server of Microsoft Exchange automatically becomes the user of the domain (primary domain server). This restricts administrators or implementers because every exchange user is a system user. Which means that the user can use the other system resources only if not restricted separately. This is not a restriction in case of Qmail, where an administrator is free to have e-mail users defined separately as well as have a single sign-on.
- In terms of security and maturity, Microsoft Exchange is still far behind the e-mail applications or solutions of Qmail.
- If carefully designed, e-mail servers such as Qmail and Postfix have superior relay controls, spam controls, customisation of solutions and so on. You will find most e-mail servers placed on the Internet based on open source rather than on closed source technology such as Microsoft Exchange.
- There are many integrated products, such as content filtering, log analysis tools, and MRTG traffic graphs, which can be associated with Microsoft Exchange server. These tools are required for customising the solution for the organisation and to make management easy for administrators. But these products are too costly to justify their procurement. Such tools and much higher integration levels can be achieved with open source solutions such as Qmail at minimal cost.

To put all this in a nutshell, if organisations want to have a scalable, high performance e-mail application server that can evolve into the future, then this is the right time to think about migration, which justifies the change. Most organisations migrating to open platforms mention cost-reduction as the major benefit. But in my view, solutions such as Qmail offer to enterprises much more than just cost-reduction. It would be better for organisations to migrate early to take the advantages, rather than wait till they are forced to do so by circumstances.

LET'S MIGRATE

As migration is a critical issue, it also needs careful planning. First, it is necessary to clearly understand the features of the Microsoft Exchange server and client used by those who have already started using this application. In my demonstration, at present, I will not touch the scheduling and calendaring features of the Exchange Server. Unless some organisation has deployed this feature, this is not going to affect the migration. Studies suggest that only one per cent of organisations use MS Exchange's calendaring and scheduling feature. Let us analyse and see how an Exchange Server is used in an organisation. Here are some common features organisations regularly use:

- Have directory service and SMTP service activated.
- POP3 service.
- Global address book.
- Internet mail connector.

- Client users mostly use Microsoft Outlook and rarely use Outlook Express.

Let us examine a complex case first. We'll assume the client is using Outlook 2000. The client's .pst file resides in the Exchange Server and it uses the global address book for accessing addresses. Our plan will target how to deploy a Qmail server and what components need integration to minimise changes to the user interface. We have to clearly understand that in our case, the e-mail client will not change and it will remain as Outlook 2000 or Outlook XP. We have to make the back-end compatible to this interface in all aspects.

- The user should be able to send and receive mails as he was doing earlier.
- The mails and the folders lying in the exchange server need to be migrated so that the user does not lose any information that was previously available.
- The global address book should be available to the user.
- Local mails should be instantaneous.

IMPLEMENTATION AND THE MIGRATION STRATEGY

- The choice of the operating system is Linux and Qmail for the e-mail application (please refer to LFY's March '03 issue for the details of implementation).
- We need to have IMAP protocol implementation on top of Qmail. We will implement courier IMAP server on Qmail (installation has already been discussed in LFY's March '03 issue).
- We need to implement Open LDAP on this Linux server to support the global address book. In my view, if we implement LDAP, then we should implement Qmail with the LDAP patch and clustering enabled. With this migration of the user's profile, the address automatically becomes available in the LDAP server and the client gets the global address book in place. Clustered Qmail with LDAP will be discussed in a later article in this series, but our initial migration can proceed without it.

As implementations of Qmail MTA and courier IMAP have already been discussed in the March '03 issue, we will now discuss the implementation of Open LDAP on Linux and creation of a database to suit our global address book.

OPEN LDAP IMPLEMENTATION ON LINUX

First we need to implement the lightweight directory access protocol (LDAP) for centralising system and network information, providing cross-platform user account databases, and even for creating a single repository of printer definitions and configuration information. Most Linux distributions come with Openldap installed. You can check it by:

```
# rpm -qa | grep openldap
openldap-devel-2.0.27-2.7.1
openldap-servers-2.0.27-2.7.1
openldap12-1.2.13-8
openldap-2.0.27-2.7.1
openldap-clients-2.0.27-2.7.1
```

Your /etc/oopenldap/slapd.conf file should have content like the following: [NOTE: as strange as it may seem the configuration file name is actually 'slapd.conf' which stands for standalone LDAP daemon]

```
include         /etc/openldap/schema/core.schema
include         /etc/openldap/schema/cosine.schema
include        /etc/openldap/schema/inetorgperson.schema
include         /etc/openldap/schema/nis.schema
database        ldbm
database        ldbm
suffix          "dc=linuxforu,dc=com"
rootpw          secret
rootdn          "cn=manager,dc=linuxforu,dc=com"
index   objectClass,uid,uidNumber,gidNumber,memberUid   eq
index   cn,mail,surname,givenname  eq,subinitial
```

In this example, suffix is the synonym for linuxforu.com, top level of the organisation is 'dc = linuxforu, dc = com', 'cn = manager' is the administrator name of the LDAP and password is 'secret'.

We need to create an actual entry for this in the LDAP directory itself using an LDIF-formatted file and 'ldapadd'. Create a file named 'test.ldif' containing:

```
dn: dc=linuxforu,dc=com
objectclass: top
```

Be sure not to leave any trailing spaces, as they may confuse 'ldapadd'.

Add to the LDAP by

```
#ldapadd -acrv -h ldap.company.com -D
"cn=manager,dc=linuxforu,dc=com" -w managers_password -f
test.ldif
```

It's pretty simple so far. Next, create an organisational unit in the directory under dc = linuxforu, dc = com to serve as a container for our address book entries.

```
dn:     ou=addressbook, dc=linuxforu, dc=com
objectClass:    top
objectClass:    organizationalUnit
ou:     addressbook
```

Add this also to the LDAP database, the same way you added the entry earlier.

Start your openldap server by /etc/rc.d/init.d/ldap start. Now, you are ready to create your global address book. You can have an entry like:

```
dn: cn=Biswajit Banerjee, ou=addressbook, dc=linuxforu, dc=com
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
cn: Biswajit Banerjee
gn: Biswajit
sn: Banerjee
mail: biswajit@linuxforu.com
ou: addressbook
telephoneNumber: 911124650722 5311
facsimileTelephoneNumber: 911162225923
mobile: 9810003018
homePhone: 911122014572
```

And similarly, add this entry to the database. There are many GUI and Web-based tools available to control and administer the LDAP. At this juncture, you must be wondering—do we need to create our global address book like this? Definitely not. We will find a method of migrating the address book into an ldif-based file and then change it according to our config and port it to the LDAP directory.

MIGRATING GLOBAL ADDRESS BOOK OF EXCHANGE TO OPEN LDAP

As far as I know, we cannot migrate the global address book directly to LDIF format. We first need to convert GAL to PAB (personal Addressbook) of Outlook.

MVP Hal Hostettler devised the following method, which requires that you have Outlook Express and the Personal Address Book (PAB) service installed in your Outlook profile:

- In Outlook, click the Address Book button to display the Address Book dialogue. Under ‘Show Names from the’, select Personal Address Book.
- To create a new PAB named GALTransfer.pab, right click on the Personal Address Book in the 'Show Names from the' list, then choose Properties. Under Path, enter a new path, if desired, and the file name GALTransfer.pab. Click on OK.
- Back in the Address Book dialogue box, under ‘Show Names from the’, select Global Address List (or any recipient's container or address book view).
- Hold down the Ctrl key as you click on each name you want in your portable GAL PAB. If you want to select all names, click on the first name in the list, then use the scroll bar to see the last name. Hold down the Shift key as you click the last name.
- Right click on the selected names, and choose 'Add to Personal Address Book' from the pop-up menu.
- Open Outlook Express. If you have names already in your OE address, you may want to export and delete them, so that they do not get mixed into your portable GAL.
- Choose File | Import | Address Book
- In the 'Address Book Import Tool' dialogue, select Microsoft Exchange Personal Address Book, and then click Import. This puts all the SMTP addresses for the GAL entries that were in the PAB into your Windows Address Book in OE.
- Return to Outlook.
- Repeat Step 2, but create a new PAB file named GALSMTP.pab this time.
- Use Outlook's File | Import and Export function to import addresses from Outlook Express into your Personal Address book.

When you finish, the GALSMTP.pab file will contain the names and SMTP addresses of everyone you selected from the GAL in Step 4. It will not, however, have any other GAL details, such as the department or phone number. If you need that information and not just the SMTP addresses, you'll need to try one of the other methods.

You can also skip steps 9-11, and instead use File | Export | Address Book in Outlook Express to export the GAL address to a comma-delimited .csv file that virtually any e-mail program should be able to import into its address book.

Once you have the .csv file any application such as Netscape can import it in ldif format (use address book import from .csv and export to ldif in Netscape Messenger). The file will look like:

```
dn: cn=ajay,mail=ajay@linuxforu.com
cn: ajay
xmozillanickname:
mail: ajay@linuxforu.com
objectclass: top
objectclass: person

dn: cn=Anup Sharma,mail=anup@linuxforu.com
cn: Anup Sharma
xmozillanickname:
mail: anup@lpsboi.com
objectclass: top
objectclass: person

dn: cn=Biswajit Banerjee,mail=biswajit@linuxforu.com
```

```
cn: Biswajit Banerjee
xmozillanickname:
mail: biswajit@linuxforu.com
objectclass: top
objectclass: person
```

This file needs modification to support the migration to our ldap database. The modified file looks like:

```
dn: cn=ajay,au=addressbook,dc=linuxforu,dc=com
objectclass: top
objectclass: person
cn: ajay
mail: ajay@linuxforu.com

dn: cn=Anup Sharma,au=addressbook,dc=linuxforu,dc=com
objectclass: top
objectclass: person
cn: Anup Sharma
mail: anup@lpsboi.com

dn: cn=Biswajit Banerjee,au=addressbook,dc=linuxforu,dc=com
objectclass: top
objectclass: person
cn: Biswajit Banerjee
mail: biswajit@linuxforu.com
```

Now you are ready to populate your database

```
#ldapadd -acrv -D "cn=manager,dc=linuxforu,dc=com" -w secret -f
addresslook.ldif
```

Next, let's configure Microsoft Outlook 2002 to use our LDAP server:

- Start Outlook and then select Tools→Email Accounts.
- Choose 'Add a new directory or address book' under the 'Directory' label and then choose Next.
- Choose 'Internet Directory Service (LDAP)' as the address book type and then choose Next.
- For 'Server Name' specify the IP address or the hostname of the LDAP server.
- Choose 'More Settings' and then select the 'Search' tab. Here you need to specify the base search path, which we specified to 'ldapsearch' using the -b option. Type ou = addressbook, dc = linuxforu, dc = com in the text field labelled 'Search base' and then choose OK.
- Choose Next.
- Outlook will present a congratulations screen. Choose Finish to close the Wizard.
- Restart Outlook to be able to use the LDAP directory you just specified.

There are two ways to test Outlook's LDAP directory access. First, let's try the fast and easy way:

- Click on the New Mail icon to bring up the New Mail window.
- In the To: field, enter 'Biswajit'. (Outlook may try to auto-complete Biswajit's name or address if you have ever e-mailed another Biswajit before. Be sure not to use this entry, as that will short-circuit the LDAP lookup.)

You can now either tab to the next field or enter Ctrl-K to force an address lookup. If you do not enter Ctrl-K, then Outlook will perform the lookup while you are doing another operation such as entering the text of the message.

Now, practically any client supporting LDAP directory services will be able to use the lookup. You can even do it with the Outlook Express.

MIGRATING MAILBOXES TO THE LINUX-BASED QMAIL SERVER

We have taken a situation where mailboxes reside on the server itself rather than on the local desktop. As the server is Windows NT and exchange is proprietary in nature, file migration will not help. For the time being, the client Outlook 2000 will need to connect to the Exchange Server with the Exchange Connector. Also, we have to create an IMAP connectivity for the client with the Qmail-courier IMAP server. Once you have configured the services on this client, you can see the personal folder containing old mails and the folders, as well as the IMAP folder typically with inbox, sent items and trash. You can drag-and-drop the personal folders into the IMAP folders. You might have to create the same folders in the IMAP too, then copy or drag the mails from the old folders to the new. Once the migration of all the mails is done, you can delete or remove the exchange connectivity from the client and observe whether the operation at the client is working properly. IMAP is an advanced protocol wherein the client can specify whether to download only the header or only the new message. But basically the mails lie on the server only and will be deleted when the client purges them. This way, the purpose or objective of keeping mails on the server is resolved.

In case you have Outlook Express or some other Internet e-mail client, migration of mailboxes is much simpler as the mails are local to the user's desktop so that no message migration is required and you can access Qmail using either IMAP or POP3.

Another migration case is of Microsoft Exchange server serving multiple location servers with a single Internet domain. This same effect is achieved using Qmail in cluster mode with the LDAP patch. Qmail performs at its best in cluster mode.

I have personally migrated exchange servers for more than 10,000 users to open source environments—using these concepts. Not only has performance improved, all the limitations that a proprietary product poses have also been minimised. Once you are able to migrate the e-mail application server to an open source solution, you'll open your organisation to a new era of enterprise e-mail computing. LFY

This article is written by Biswajit Banerjee, Director, Tetra Information Services Pvt Ltd, which is into Linux corporate solutions. He can be contacted at biswajit@tetrain.com