Implementing POP3 and IMAP4 Using Dovecot

AfNOG 2013
Scalable Internet Services (SS-E)
Lusaka, Zambia
Presented by Michuki Mwangi
(Built on materials developed by Joel Jaeggli)

What is POP3

- POP3 stands for Post Office Protocol ver 3

- Described in RFC1913

- Runs on TCP Port 110 as a client server function

- Allows for a maildrop service (similar to the post box mail service) hence the name

- By design its limited in features to download and delete email from server

- Security was also limited to using APOP (md5 hash for authentication

- RFC 2449 proposed POP3 extensions which included SASL Mechanism, Expiry, Pipelining, etc.

- RFC 2595 describes using TLS with POP3 also known as POP3s and runs on port 995

What is IMAP4?

- Internet Message Access Protocol version 4

- Described in RFC 1730

- Runs on TCP Port 143 as client-server function

- More advanced in features compared to POP3

- IMAP4 stores mail on server and copies can be transferred to the client on request.

- By default only the message headers are sent to the client, the rest of the message is accessed on opening the email.

- Allows client to access and manipulate email residing on a server, creation of folders, filters, etc.

- RFC 1731 describes the IMAP Authentication Mechanisms

- RFC 2595 describes using TLS with IMAP4 running on TCP port 993

Mail Storage Formats

- Mailbox Format (Mbox)

- Defined in RFC 4155

- All messages in an Mbox mailbox are concatenated and stored as a plain text in a single file

- Mails are stored in RFC822 format with a blank space separating each message (2 spaces as each message has one space) and "From" determining start of next message.

- Mbox has a distinct disadvantage in cases of large mailbox (a single large file) requires more resources to read/open and can be slow depending on the servers load.

Maildir Storage Format
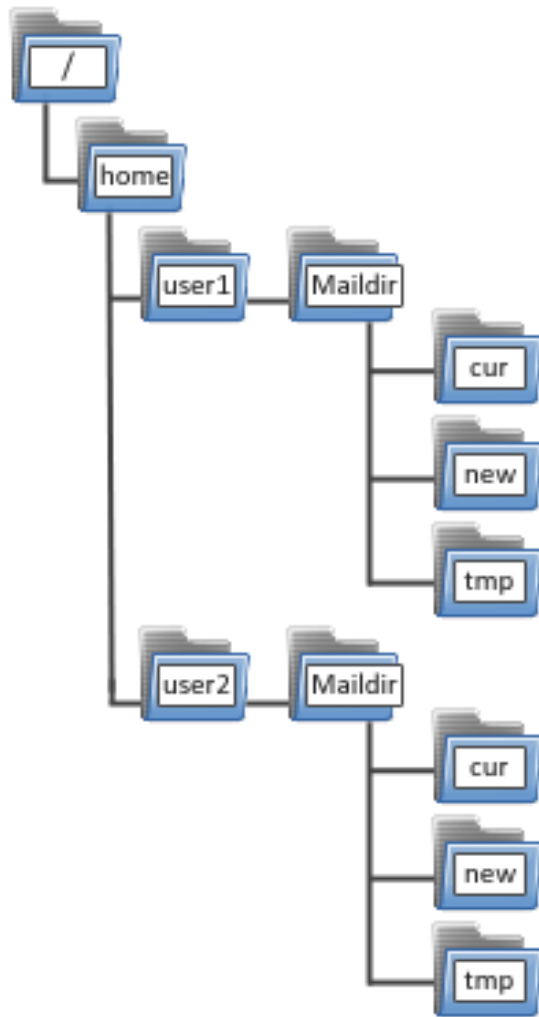
- Mail Directory Format (Maildir)

- Each message is stored in a separate file with a unique name and each folder in a directory

- Maildir++ provides extension to the Maildir specification providing support for subfolders and quotas.

- Maildir directory has 3 folders **temp**, **new** and **current**

How Maildir Works

- The mail delivery agent stores all new emails to the mailbox in the tmp directory with a unique filename. (unique = time + hostname+ random generated number)

- The MDA creates a hard link to the file in tmp/ unique to new/unique

- The Mail User Agent will check for new emails in new folder and move them to current folder

- The MUA modifies the filename to add a colon (:), a '2' and various flags to represent message status i.e read, replied, forwarded, deleted, etc

-

What is Dovecot?

- High-performance POP and IMAP server

- Developed byTimo Sirainen

- Unlike say UW IMAP it wasn't written in the 80s

- Transparently index's mailbox contents (Why is this important?)

- Supports both mbox and maildir formats

- Capable of operating in an environment with minimal locking. (Why is this important)

- Graceful around failures (index repair for example)

- Designed with Security in mind – support for Authentication Mechanism and SSL/TLS

Let's install it the FreeBSD way
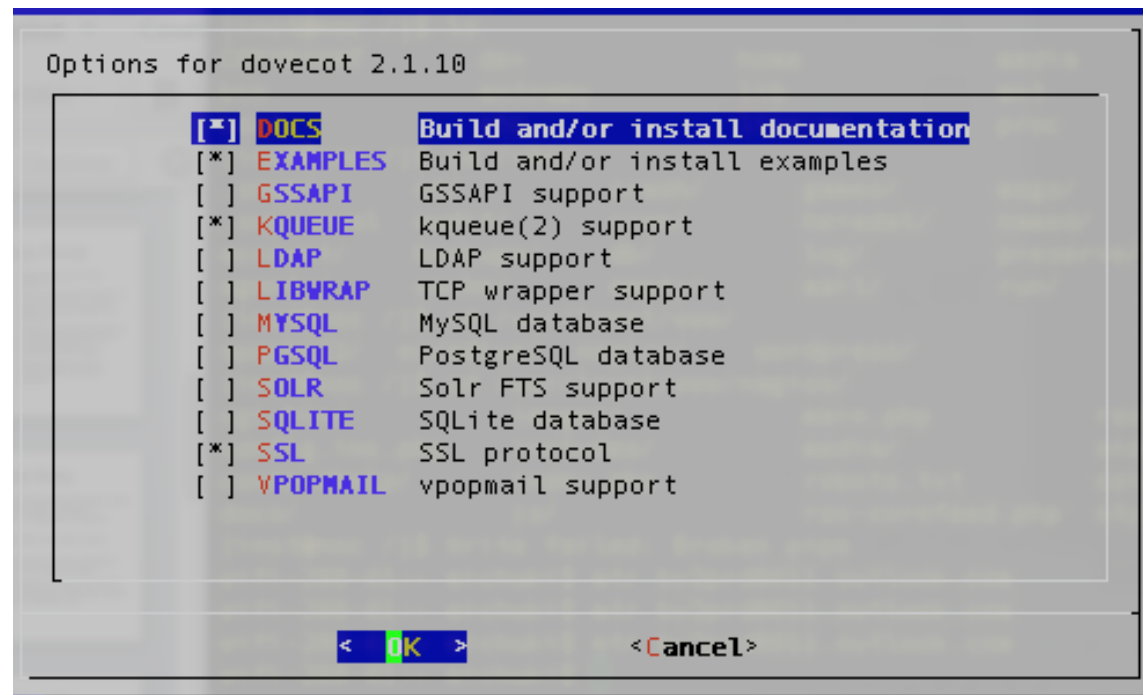
- `#cd /usr/ports/mail/dovecot2`

- `#make install clean`

- Note all the options for databases!

  - It is typical for small applications to do authentication of users using the unix password file or PAM.

  - Big mail installations can use an SQL database interface for the storage of user credentials.

- Select the Mysql Option for this exercise

```
Options for dovecot 2.1.10

     [*] DOCS       Build and/or install documentation
     [*] EXAMPLES   Build and/or install examples
     [ ] GSSAPI     GSSAPI support
     [*] KQUEUE     kqueue(2) support
     [ ] LDAP       LDAP support
     [ ] LIBWRAP    TCP wrapper support
     [ ] MYSQL      MySQL database
     [ ] PGSQL      PostgreSQL database
     [ ] SOLR       Solr FTS support
     [ ] SQLITE     SQLite database
     [*] SSL        SSL protocol
     [ ] VPOPMAIL   vpopmail support


          <  OK  >            <Cancel>
```

FreeBSD install cont

- Take a look at;

  ```
  $ less /usr/local/etc/rc.d/dovecot
  ```

- Enable Dovecot by editing /etc/rc.conf and adding the following line at the end of the file.

  - ```
    dovecot_enable="YES"
    ```

- Ok, now we could start it but we really need to configure it first.

- Copy the default Dovecot Config file from the Examples

  - ```
    $ cp -r /usr/local/share/doc/dovecot/
    example-config/*  /usr/local/etc/dovecot/
    ```

Dovecot 2 Protocols Configuration

- Open and edit the dovecot config file

  - `$ vi /usr/local/etc/dovecot/dovecot.conf`
- Note that the default listening services are:

  `protocols = imap pop3 lmtp`
    - *The TCP listeners are on 110, 143, 993, and 995*
    - *If you need the unencrypted versions of the protocol for some reason (e.g. a webmail application) then you should firewall them off from the rest of your end users (end-user clients should never be be allowed to connect insecurely)*
- If you have working SSL Certificate (from Apache-SSL session), uncomment and add imaps and pop3s protocols as follows;

  `protocols = imap imaps pop3 pop3s lmtp`

- If you do NOT have working SSL Certificates, uncomment and retain the imap and pop3 as follows;

  `protocols = imap pop3 lmtp`

Dovecot 2 SSL Configuration

- ***If you do NOT have a working SSL Certificate, follow the next 3 steps***

1. ***1. Edit the file /usr/local/etc/dovecot/conf.d/10-ssl.conf and find line***

    # ssl = yes

2. ***2. Uncomment the line and modify it to NO***

    ssl = no

    3. Comment the following lines

    - #ssl_cert = </etc/ssl/certs/dovecot.pem
    - #ssl_key = </etc/ssl/private/dovecot.pem

- **If you have SSL Certs Working during Apache Session,** *edit the file /usr/local/etc/ dovecot/conf.d/10-ssl.conf and find lines*

    - ssl_cert = </etc/ssl/certs/dovecot.pem
    - ssl_key = </etc/ssl/private/dovecot.pem

- ***MODIFY*** above lines and set ***PATH*** to point at the certificate and keyfile that was created during the apache tutorial. i.e

    - ssl_cert = </usr/local/etc/apache22/server.crt
    - ssl_key = </usr/local/etc/apache22/server.key

- Save and close the 10-ssl.conf file

Dovecot 2 Authentication Config

- ***Edit the file /usr/local/etc/dovecot/conf.d/10-auth.conf***

- Disable plaintext authentication by finding the line below

  `#disable_plaintext_auth = no`

- Uncomment the line and Set the value to yes as below

  `disable_plaintext_auth = yes`

  - Note: unencrypted connections can still be made from localhost!

Dovecot 2 Mailbox Configuration

- The mail storage by Exim is in /home/%u/mail in Maildir format

- The default Dovecot mailbox and storage is not defined in Dovecot 2

- To define the Mailbox location and format, edit the file /usr/local/etc/dovecot/conf.d/10-mail.conf

- Locate the line:

  `#mail_location =`

- Uncomment and add the maildir format and location

  - `mail_location = maildir:~/mail/`

- Ok we should have a sufficiently tuned dovecot to be able to start it.

- *`/usr/local/etc/rc.d/dovecot start`*

Done

- If everything works correctly you should be able to point an imap client towards your system at port 993 or pop3 clients on port 110

- Alternatively; using telnet

  # telnet localhost 110

  user afnog

  pass afnog

  list

  quit