DOVECOT

Timo Sirainen
Dovecot Solutions Ltd
http://www.dovecot.org/

The Talk

- Introduction
- v2.0: doveadm, sis, dsync, director
- v2.1: imapc, sql, nosql mail backends
- Questions

Dovecot?

- IMAP, POP3 mail server
- Mail delivery agent + LMTP
- Sieve mail filtering language + ManageSieve
- (No SMTP. Postfix/Exim is fine)

Dovecot History

- Written mostly by me
  - Started in summer of 2002
- Pigeonhole Sieve by Stephan Bosch
  - Started 2006-2007
- v2.0 in late 2010
  - Finally highly modular!
- Slowly nearing perfection ☺
- Dovecot Solutions company created in April 2011

Dovecot Features

- High performance (low disk I/O usage)
- Highly configurable/flexible
- Easy migration from other servers
- Admin-friendly
  - All errors are logged
    - Error log should stay empty!
  - Understandable error messages
  - Automated fixing of (corruption) errors

doveadm

- All admin commands are being added to it
- Supports plugins (already quota, acl)
- Read/modify/debug mailboxes
- passdb/userdb lookups
- List/kick user connections
- Lots more

doveadm: Managing mailboxes

- dbox: grep/mv/rm no longer possible
- force-resync INBOX
- expunge mailbox Trash savedsince 7d
- mailbox delete Trash/foo
- move Trash mailbox INBOX from bob@example.com

doveadm: scripting

```
doveadm search -u bob mailbox work/* subject todo unseen |
while read guid uid; do
  doveadm fetch -u bob body mailbox-guid $guid uid $uid > msg.$guid
done
```

doveadm: debugging

- acl debug
- auth, user
- dump
- log find

doveadm: Status

- who
  - kick <user>/<ip>
- proxy list
  - proxy kick <user>
- quota get
  - quota recalc

Single Instance Storage

- Supported by dbox (single, multi)
- mail_attachment_dir points to external shared attachments dir
- mail_attachment_min_size + plugins determine what are attachments
- Attachments are written via “lib-fs”
- Simplest lib-fs backend: **posix**
  - Useful if your filesystem already does SIS.

SIS

- **sis** backend: deduplicate immediately
- **sis-queue** backend: deduplicate later using “doveadm sis deduplicate”
- Both backends currently do byte-by-byte comparison before deduplication
- Potential matches are found via hashes
  - Configured by mail_attachment_hash
  - e.g. sha1, sha256, file size

dsync

- Local and remote (e.g. via ssh)
  - dsync -u username mirror maildir:/backup/user/Maildir
  - dsync -u username backup ssh -i id_dsa.backup mailuser@example.com dsync -u username
- Backup (makes dest look exactly like src)
- Two-way non-destructive mirroring
- Mailboxes always in consistent state
  - Unlike e.g. rsync from a live filesystem

dsync: Two-way Mirroring

- Mailboxes on both sides can be modified without losing any changes
  - Requires both sides to run Dovecot v2.0 to be reliable
- Online migrating between mailbox formats
- Online moving mailbox between storages
- Poor man’s replication

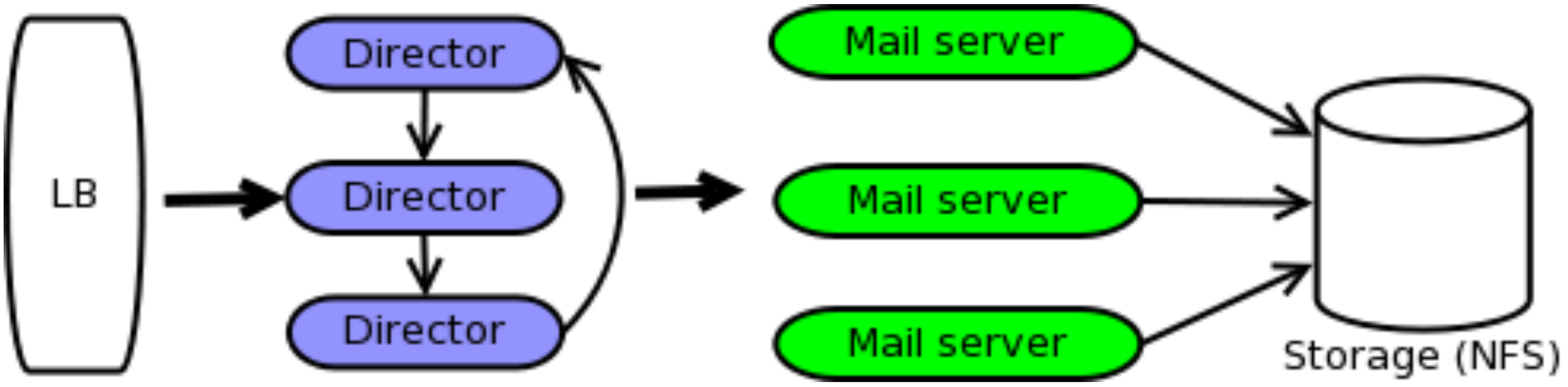
dsync: Online migration

- Example: Maildir -> mdbox
- Initial (slow) dsync
  - dsync -u username mirror mdbox:~/mdbox
- If it took a long time, maybe run it again
- Change userdb to return mail=mdbox for user
- Kill user’s all connections (doveadm kick)
- Final dsync
  - dsync -u username mirror maildir:~/Maildir
- Delete ~/Maildir

dsync: Poor man’s replication

- Multi-master!
- Modify any of the masters
- Run dsync to synchronize masters
  - e.g. every 5 minutes in cronjob..
  - Create a plugin to run dsync when needed
- dsync works quite well with high latency

Director Proxy

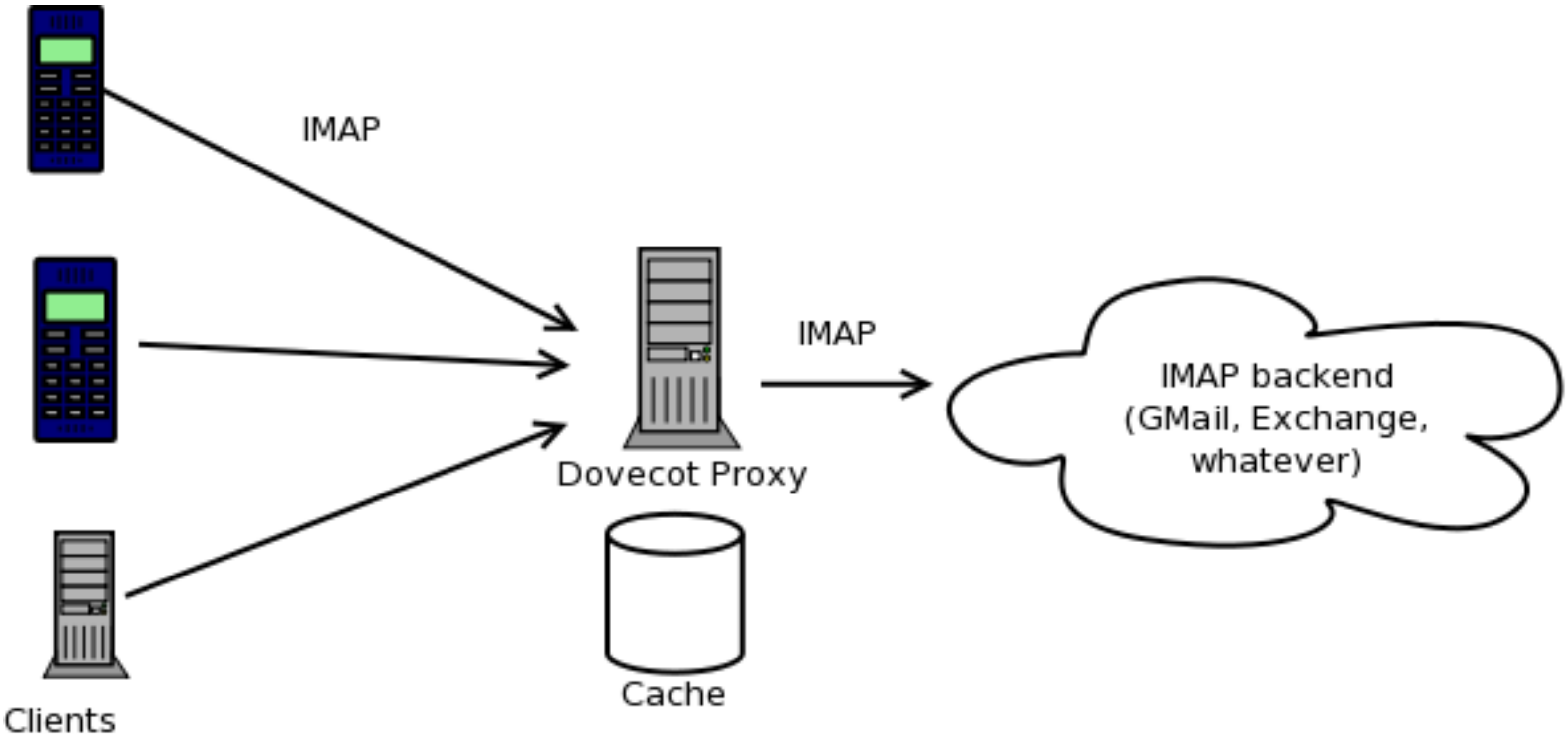


Director Proxy

- Assumes shared disk storage across mail servers
- Temporary user → mail server mapping
- Guarantees that user won’t be simultaneously accessed by 2 mail servers
- Fixes all NFS caching troubles
- Better performance with clusterfs
- Shared mailboxes are currently trouble..

v2.1: IMAP Accelerator(?)

- Naming?.. (“proxy”, “imapc backend”)



IMAP Accelerator(?)

- Remote IMAP server is just another mail storage backend (like Maildir, mbox, SQL, etc.): “imapc”
- Kind of like a “smart IMAP proxy” (but also for POP3, LMTP, doveadm, whatever)
- Many new use cases for Dovecot!

Caching IMAP Proxy

- To reduce load and/or disk I/O and/or latency
- 1+ caching proxies in front of real server
- Kind of like master/slave load balancing

Fixer Proxy

- Dovecot is 100% IMAP RFC compliant (AFAIK)
- If client A doesn’t work with server B, user can:
  - Try to convince client developer to fix it
  - Try to convince server developer to fix it
  - Switch client
- A new option: Dovecot proxy!

Security Proxy

- Public IMAP not allowed by some organizations due to security concerns
- No VPN in phone? → No mobile email
- Dovecot can act as a secure mail gateway
  - Strict privilege separation: one connection per process, chrooted
  - SELinux further restricting

Filtering Proxy

- Modify messages on the fly
- Examples:
  - Easy PGP support to all clients
    - Only Dovecot proxy needs to be trusted with keys
  - Content recoding
    - Pictures/videos for mobile clients
    - Plaintext-only for your sailing/jungle trip with satphone

Perfect Migration Tool

- dsync + imapc = perfect migration from any IMAP server to Dovecot
  - IMAP UIDVALIDITY, UIDs, MODSEQs
  - Avoids IMAP clients redownloading mails

How does the proxy work?

- Uses only simple commands currently → works fine with buggy servers
  - FETCH BODY[HEADER], BODY[], UID, FLAGS, INTERNALDATE
- Maybe optionally more complex ones in future
  - SEARCH, SORT, THREAD, …

How does the proxy work?

- Supports Dovecot index/cache files
  - No refetching cached content
- Often high latency → command pipelining / mail prefetching
  - mail_prefetch_count setting
    - Linux: Readahead with Maildir/sdbox also
  - In future: Process more commands while waiting for remote command replies

Key-value Storage

- Key-value databases / cloud filesystems are often high latency
- imapc pipelining/prefetching code can be reused
- → Dovecot should soon have efficient support for key-value databases.

Key-value Storage

- Probably based on sdbox/mdbox mailbox format, modified to use “lib-fs”
- lib-fs is a simplified filesystem access API
  - Atomic (re)create/append, no overwriting
  - No locking
  - Easy to add support for many kinds of db
- Assume server is master, but support application side conflict resolution (if db supports)

SQL Storage

- Dovecot mail storage access API is very flexible, powerful and easy to support
- Already written code for proprietary storages:
  - Read-only INBOX from MySQL: 2 hours of work
  - Read-write hybrid filesystem + MySQL
- Maybe a generic PostgreSQL/MySQL backend in future

Questions?

Picture by *Cyril Thomas*