Table of Contents

IndiMail

1.1. Introduction

Electronic mail is an asynchronous messaging technology. The person that you are trying to reach does not have to be available that instant. This fact makes Electronic mail very convenient. Hence it is not surprizing that the killer app even today is the electronic mail.

There are many types of electronic mail systems in the world today. Most of them never learned to share (or did not want to) - Products from Microsoft, Domino, etc. Most major computer vendors offer a proprietary email systems. IBM, Digital Equipment Corporation, Microsoft, Hewlett-Packard, and others have their own ways of performing this simple task in complicated ways (primary motivation being commercial – *more complicated the better it is for the vendor*).

The defacto standard of Internet mail system has now been based on the SMTP protocol (Simple Mail Transport Protocol).

1.2. Internet Email

The internet mail system can be easily conceptualized by introducing the key elements. A *message* is a piece of information that one Internet user wishes to send to another. It may include multiple parts, including binary files, that may be attached to

the message. A message is send from an MUA; MUAs are also used to read messages that are received from others.

The most important elements of the Internet mail systems are:

- Mail user agent (MUA)
  A client program used by a user to send or receive email. An MUA could also be a program or script that emulates the behaviour of a typical MUA by sending or receiving email. e.g. Sylpheed, Mutt, Pine, Outlook Express.
- Mail transfer agent (MTA)
  A server program that transfers email from one machine on the Internet to another – a mail server. e.g. sendmail, qmail, postfix.
- Mail delivery agent (MDA)
  A small program used by a MTA to write a message into a user's mailbox. e.g. sendmail, maildrop, local delivery agent for qmail, postfix.
- Mail retrieval agent (MRA)
  A service that retrieves messages from a mailbox on a remote server to a user's MUA. e.g. IMAP, POP3 agents

The internet mail system appears to be rather complex. In reality, it is a system of components, each of which does a simple job. The devil, of course, is in the details. This document is all about IndiMail – What benefits it provides you and how you can use it for your needs.

IndiMail is an integrated solution of the following packages qmail, serialmail, qmailanalog, dotforward, fastforward, mess822, daemontools, ucspi-tcp, Indimail - Management of Virtual domains, Courier IMAP/POP3, Bogofilter - A Bayesian Spam Filter, Fetchmail, other useful utilities (pack, unpack, altermime, ripmime, flash). IndiMail also comes with the famous fortune program – resurrected just for you.

The MTA is based on ***qmail*** a trusted, reliable and secure SMTP server which takes care of most of the drawbacks of existing SMTP servers and also adds lot of new features.

I have seen IndiMail easily handle 8,000,000+ users and can do close to 6 million deliveries/day, and host 5000+ domains

1.3. IndiMail Feature list

IndiMail is a secure, reliable, efficient, simple mail server with all major components coded entirely in C. It has a extremely small footprint. It provides the functionality of delivering mails to the User's mailbox and retrieving the same by any third party MUA used by the Internet Community.

However IndiMail does not provide any Email Client or web mail Client. It is designed for typical Internet-connected UNIX hosts and this is what it is -
This section has been heavily borrowed from The qmail web page, http://cr.yp.to/qmail.html and Dave Sill's Life with qmail.

The feature list can also be read at
http://groups.google.co.in/group/indimail/web/detailed-feature

Secure: Large part of IndiMail's security comes from qmail. Different sub-systems use different userids to create partitions which prevent one system accessing the other.

Reliable: Since indimail uses qmail as its MTA a message, once accepted into the system, will never be lost. IndiMail also optionally supports Maildir, a new, super-reliable user mailbox format. Maildir, unlike mbox files and mh folders, won't be corrupted if the system crashes during delivery. Even better, not only can a user safely read his mail over NFS, but any number of NFS clients can deliver mail to him at the same time.

Efficient: qmail's modular, lightweight design and sensible queue management coupled with IndiMail's queue parallelism make it the fastest available message transfer agent.

Simple: IndiMail is vastly smaller than any other Internet mail systems and hence we have better control on its behaviour. Some reasons why:

1. IndiMail has one simple forwarding mechanism that lets users handle their own mailing lists.
2. IndiMail is instantly triggered by new items in the queue, so the IndiMail system has just one delivery mode: fast+queued.
3. Total control on the processing load and behaviour with few control files or environment.

IndiMail borrows Mailing list management which is one of qmail's strengths. Notable features:

- qmail lets each user handle his own mailing lists. The delivery instructions for user-whatever go into ~user/.qmail-whatever.

- qmail makes it really easy to set up mailing list owners. If the user touches ~user/.qmail-whatever-owner, all bounces will come back to him.
- qmail supports VERPs, which permit completely reliable automated bounce handling for mailing lists of any size.

SPEED:

- IndiMail blasts through mailing lists two orders of magnitude faster than sendmail.
- qmail automatically prevents mailing list loops, even across hosts.
- qmail allows inconceivably gigantic mailing lists. No random limits.
- qmail handles aliasing and forwarding with the same simple mechanism. For example, Postmaster is controlled by ~alias/.qmail-postmaster. This means that cross-host loop detection also applies to aliases.

Setup:

- automatic adaptation to your UNIX variant---no porting needed
- Linux, SunOS, Solaris, and more
- automatic per-host configuration (configure, config, config-fast), make install.
- quick installation---no big list of decisions to make
- High degree of automation of configuration through **svctool**.

Security:

- clear separation between addresses, files, and programs
- minimization of setuid code (qmail-queue)
- minimization of root code (qmail-start, qmail-lspawn)
- five-way trust partitioning---security in depth
- optional logging of one-way hashes, entire contents, etc. (EXTRAQUEUE)
- virus scanning through qscanq

Message construction:

- RFC 822, RFC 1123
- full support for address groups
- automatic conversion of old-style address lists to RFC 822 format
- sendmail hook for compatibility with current user agents
- header line length limited only by memory
- host masquerading (control/defaulthost)
- user masquerading ($MAILUSER, $MAILHOST)
- automatic Mail-Followup-To creation ($QMAILMFTFILE)

SMTP service:

- **RFC 2821**, RFC 1123, RFC 1651, RFC 1652, RFC 1854, **RFC 1870**, **RFC 1893**
- 8-bit clean
- 931/1413/ident/TAP callback
- relay control---stop unauthorized relaying by outsiders **(control/rcpthosts)**
- no interference between relay control and aliases
- automatic recognition of local IP addresses
- per-buffer timeouts
- hop counting
- parallelism limit (tcpserver)
- per host limit (tcpserver - MAXPERIP)
- refusal of connections from known abusers (tcpserver, badmailfrom, badmailpatterns, badhelo, blackholedsender, blackholedpatterns)
- STARTTLS, TLS/SSL extension
- goodrcptto, goodrcptpatterns which override the above
- blackholercpt, blackholercptpatterns for blackholing mails to specific senders.
- Control files spamignore, blackholedsender, badmailfrom, relaymailfrom, badrcptto, chkrcptdomains, goodrcptto, blackholercpt can be specified in cdb format as well as stored in MySQL tables.
- relaying and message rewriting for authorized clients.
- authenticated SMTP PLAIN, LOGIN, CRAM-MD5 (HMAC (rfc1321, rfc2104, RFC 2554))
- POP/IMAP before SMTP
- ETRN (RFC 1985)
- ODMR (RFC 2645)
- RBL/ORBS support **(rblsmtpd)**
- SPAM Control (Reject/Tag/Accept) using Bayesian techniques
- Per User control of Environment variables via ***envrules***
- High Performance MS Virus Control via control file ***viruscheck*** and control file ***signatures***
- Content Filtering and blocking of prohibited attachments via control file ***bodycheck***
- Ability to bounce mails for unknown/inactive users (CHECKRECIPIENT), ability to have the check for selective domains using control file ***chkrcptdomains***
- Antispoofing mode (turned on by environment variable ANTISPOOFING)
- Masquerading ability.
- Multiline greetings via control file ***smtpgreeting***
- Message Submission Agent – **MSA** (RFC 2476)
- Domain IP address pair access control via control file ***hostaccess***
- Per User accesslist via control file ***accesslist***
- **SPF** – Sender Permitted From
- Per User control of environment variable by ***envrules*** (rules file set by

environment variable FROMRULES)

- Greylisting capability using qmail-greyd / greydaemon.
- Bounce Address Tag Validation (BATV)

Queue management:

- instant handling of messages added to queue
- parallelism limit **(control/concurrencyremote, control/concurrencylocal**)
- split queue directory---no slowdown when queue gets big
- quadratic retry schedule---old messages tried less often
- independent message retry schedules
- automatic safe queuing---no loss of mail if system crashes
- automatic per-recipient checkpointing
- automatic queue cleanups (**qmail-clean**)
- queue viewing **(qmail-qstat)**
- detailed delivery statistics (qmailanalog)
- Configurable number of queues and time slicing algorithm for load balancing via **qmail-multi.**

  A queue in indimail is configurable by three environment variables QUEUE_BASE, QUEUE_COUNT, and QUEUE_START. A queue in IndiMail is a collection of queues. Each queue in the collection can have one or more SMTP listener but a single delivery (qmail-send) process. It is possible to have the entire queue collection without a delivery process (e.g. SMTP on port 366 – ODMR). The QUEUE_COUNT can be defined based on how powerful your host is (IO bandwidth, etc). NOTE: You do not require multiple installation of qmail to achive configurable number of queues.

- Ability to hold local, remote or both deliveries (***holdlocal***, ***holdremote*** control file)
- Qmail Queue Extra Header – Ability to pass extra headers to local and remote deliveries via qmail-queue (Environment variable QQEH).
- External Virus scanning via **QHPSI** – Qmail High Performance Scanner Interface
- Ability to extend QHPSI interface through plugins. The keyword **plugin:*shared_lib*** defined in the environment variable QHPSI denotes '***shared_lib***' to be loaded.
- Virus scanner **qscanq**. Ability to detect virus via a third party scanner defined by SCANCMD environment variable (clamscan, clamdscan, etc)
- Blocking of prohibited filename extensions via **qscanq** program
- DOMAIN KEYS (**qmail-dk**) RFC-4870
- Domainkey Identified Mail – DKIM (qmail-dkim), with Author Domain Signing Practice (ADSP) and Sender Signing Practice (SSP) RFC-4871
- Set all header values listed in ***envheader*** control file as environment variables.
- Log all headers listed in control file ***logheaders*** to stderr.
- Remove all headers listed in control file ***removeheaders*** from email.
- Ability to do line processing instead of block processing.

- qmail-nullqueue – blackhole the mail silently.

Bounces:

- QSBMF bounce messages---both machine-readable and human-readable
- HCMSSC support---language-independent RFC 1893 error codes
- double bounces sent to postmaster
- Ability to discard double bounces
- Ability to preserve MIME format when bouncing.
- Control of bounce process via ***envrules*** (rules file controlled by environment variable BOUNCERULES)
- limit size of bounce using control file ***bouncemaxbytes***
- **External bounce processor by setting environment variable BOUNCEPROCESSOR**

Routing by domain:

- any number of names for local host **(control/locals)**
- any number of virtual domains **(control/virtualdomains**)
- domain wildcards (**control/virtualdomains)**
- configurable percent hack support (**control/percenthack**)
- **Clustered Domain**. Same virtual domain can exist on multiple hosts, each having its own set of users. Provides Load Balancing

Remote SMTP delivery:

- **RFC 2821**, RFC 974, RFC 1123, **RFC 1870**
- 8-bit clean
- automatic downed host backoffs
- Configurable tcp timeouts for downed host backoffs.
- automatic switchover to next best MX
- artificial routing---smarthost, localnet, mailertable **(control/smtproutes)**
- Static and Dynamic Routing. (SMTPROUTES environment variable)
- user location aware routing. Allows users in a domain to be distributed across multiple hosts.
- Support for jumbo ISP (**control/smtproutes.cdb**)
- per-buffer timeouts
- STARTLS, TLS extension
- passive SMTP queue---perfect for SLIP/PPP (serialmail)
- AutoTURN support (serialmail)
- Authenticated SMTP (userid/passwd in control/smtproutes)
- Spam control (SPAMFILTER environment variable)
- Environment variable control via ***envrules*** **(rules file controled by environment variable RCPTRULES)**

- **QMAILREMOTE environment variable to run any executable/script instead of qmail-remote**

Local delivery:

- user-controlled address hierarchy : fred controls fred-anything
- mbox delivery
- reliable NFS delivery (maildir)
- user-controlled program delivery: procmail etc. (qmail-command)
- optional new-mail notification (qbiff)
- detailed Delivered-To Headers
- optional NRUDT return receipts (qreceipt)
- autoresponder rfc3834 compliance (provide Auto-Submitted, In-Reply-To, References fields (RFC 3834))
- conditional filtering (condredirect, bouncesaying, vfilter)
- **Environment variable control via *envrules* (rules file controled by environment variable RCPTRULES)**
- Eliminate duplicate messages
- **QMAILLOCAL environment variable to run any executable/script instead of qmail-local**

Other:

- **Change concurrency of tcpserver without restart.**
- **SSL/TLS for encryption.**
- **Ability to restrict connection per IP (MAXPERIP)**
- **run shutdown script if present on svc -d**
- **ability to log svscan output using multilog**
- **inlookup – High Performance User Lookup Daemon.**
- **indisrvr – Indimail Administration Daemon.**
- **spawn-filter - Ability to add disclaimer, run multiple filters before local/remote delivery.**
- **Proxy for IMAP/POP3 Protocol**
- **svctool – Configuration tool for IndiMail.**

Colour Codes

	Features specific to IndiMail / significantly different from what is existing on public domain
	Features borrowed from patches/code available on internet

In addition to all the above mentioned features IndiMail has implemented a host of new and creative features to increase the reliability, efficiency and speed.

1. 100% guarantee of service uptime.

Automated system for continuous monitoring of all critical mail services without using resources on the system. System brings up the service in case it goes down in the rarest of cases.

2. **Increased Queue Efficiency**
   Highly optimized queue writing and reading by implementing time sliced multiplexing and a queue collection of multiple qmail queues. These queue system enables IndiMail to do around 3 Million plus deliveries/day for an ISP in India.

3. **Support for most of the mail clients**

   1. Evolution (good GUI with groupware)
   2. Mozilla
   3. Thunderbird
   4. Netscape
   5. Eudora
   6. Pine (Cute and simple)
   7. MUTT (so far the best email client, with the most configurable features, downside - no gui)
   8. Sylpheed (A real fast client based on GTK)
   9. balsa
   10. Outlook Express (IMHO dumb, buggy, piggy, virus prone; use it at your own risk)
   11. Microsoft Outlook (IMHO dumb, buggy, piggy, virus prone; use it at your own risk)
   12. sqwebmail (Works by accessing the filesystem)
   13. squirrelmail (A good configurable web client. Accesses mail via IMAP)
   14. Any Client supporting IMAP/POP3 (including Web Based)

4. **Close to the Operating System**
   Entire code written in pure C with NO modern day pollutants (java, perl, python etc), ensuring application's closeness to the OS for optimized healthy performance.

5. **Shared library**
   Entire APIs in a shared library (**libindimail.so**) enabling upgrades and patch fixes without bringing down production systems or replacement of binaries.

6. **Support for very large no of Domains.**
   Support for 1 to 23 million virtual email domains using a "grow as it goes" balanced directory tree.

7. **Users Per Domain**
   Support for 1 to 23 million email users per domain using the same balanced tree

structure.

8. Pluggable Authentication modules.

IndiMail uses checkpassword authentication mechanism.

9. Relay control

The Application supports authenticated SMTP and POP before SMTP to prevent unauthorized relaying.

10.Inbuilt firewall.

Prevents unauthorized hosts from accessing IndiMail services using tcpserver. Per IP concurrency limit prevents DOS attacks.

11.Automated password generation.

Facility for users to automatically generate passwords.

12.SPAM Control

1) Non acceptance of mails from known chronic spammers based on the sender email id or the pattern of his id including the domain
2) Tarpitting - Throttle mails from Potential spammers sending huge no of emails.
3) Per IP concurrency limit (to prevent potential denial of service attack from one IP)
4) vfilter – filter can be additionally used to control spam through vfilter facility.
5) Automatic building up of ***badmailfrom/badrcptto/spamdb*** by using **chowkidar** in **cron(1)**
6) Chronic spammers can also be blackholed through the control file ***blackholedsenders.***
7) Bulk senders/spammers can be forwarded to a slower queue or any other queue via the control file ***envrules***.
8) Host access control via ***hostaccess*** control file.
9) Content-filtering via ***bodycheck*** control file and virus filtering via ***viruscheck*** control file.
10) RFC-2505, RFC-2635
11) Bayesian spam filter using bogofilter. Any external spam filter configurable by setting SPAMFILTER environment variable.
12) Ready to use SPAM database corpus

13.Runtime configurable mail size limit

1. Domain wide mail size configured via control file databytes.
2. User Specific mail size limit for local users via ***envrules***.

14.Accounting

1. last login/last authentication
2. Last password change
3. User addition date/time

15.Statistics and Reports (Optional)

1. Total mails received/sent per day
2. Mails sent/received per user/Domain/IP
3. User additions per day

16.Alerts (Optional)

Email alerts to administrators for system abnormalities, queue status, system load..

17.Autoresponder Facility

Autoresponder facility which sends customized automated replies. A typical example of this is vacation.

18. Filtering Facility

This feature can be used to filter messages and take various action on a match. The match could be any criteria based on the standard mail headers. The action on match can be rejection of mail, delivery to a specified folder or forward to another email address (**vfilter**).

19. Bulletin Feature

This feature (**vbulletin**) delivers instant messages to all users using a single i-node and hence conserving disk space and disk IO. ***No other system in milky way has this feature***.

20. Configurable logging (Optional)

Logging can be based on real world admin's comments and requirements.

21. Stale Account Management

Optional facility to automatically delete the stale accounts (**vdeloldusers**)

22.IMAP4 Rev1 and POP-3 Support

Provides highly configurable out of the box IMAP and POP-3 Support.

23.Mail Cluster

This provides clustered mailing functionality or a mail cluster (and in fact it also works with any proprietary solutions supporting SMTP). Clients can get value add if they have legacy mail servers supporting SMTP and want move out some users from legacy solutions to the Open scalable solution - IndiMail, without changing the domain name. Clients can also get out of existing legacy mailing solutions by deploying the relay server software without causing customer impact and

downtimes. The relay server can provide clustering solution for a legacy system where clustering is not supported.

1.4. Related packages

IndiMail like qmail follows the classic UNIX philosophy that each tool should perform a single, well-defined function, and complex functions should be built by connecting a series of simple tools into a "pipeline". Many standard UNIX utilities can also be plugged into IndiMail.

- ucspi-tcp--an inetd replacement
- fetchmail-- tools for mail retreival.
- ezmlm--a mailing list manager for qmail
- bogofilter--a fast and powerful bayesian spam filter.
- Clamav – Open Source Virus Scanner
- altermime, ripmime – For MIME processing

1.5. Documentation

1.5.1. man pages

The IndiMail distribution comes with a complete set of man pages. After installation, they're in /var/indimail/man. You'll probably need to add that directory to your MANPATH environment variable.

Shell	Command
Bourne (/bin/sh)	MANPATH=$MANPATH:/var/indimail/man; export MANPATH
bash, Korn	export MANPATH=$MANPATH:/var/indimail/man
C Shell	setenv MANPATH $MANPATH:/var/indimail/man

At this point, commands in the format "man name-of-indimail-man-page" should display the appropriate man page.

Note: The man pages are loaded with information, but they require careful reading because they're written in a very dense, technical style. You might want to print off a set and read them through once to familiarize yourself with what's there and where it is. Very little information is repeated on multiple pages, so if you don't know where something is covered, it can be hard to find it.

1.5.2. Docs

The qmail distribution includes a series of documents that are installed under /var/indimail/doc. They include:

- FAQ: Frequently Asked Questions, with answers
- INSTALL*: Installation documentation
- PIC.*: Descriptions of how qmail performs key tasks. See the Architecture appendix for more information.
- Various other installation-related documentation

These docs are also available on-line from
http://groups.google.co.in/group/indimail

1.5.3. FAQs

There are two official FAQ (Frequently Asked Questions, with answers) documents:
- /var/indimail/doc/FAQ, the plain text version, for qmail, and
- /var/indimail/doc/FAQ.pdf, the pdf version, for IndiMail, and
- The web FAQ

1.5.4. Other Web Sites

http://www.indimail.org
http://cr.yp.to/qmail.html: the official qmail home page.
http://www.qmail.org: the unofficial qmail home page. Contains lots of information about add-ons and patches, and links to many good qmail web pages on other sites.

1.6. Mailing Lists

There are four Mailing Lists for IndiMail

1. indimail-support - You can subscribe for Support at https://lists.sourceforge.net/lists/listinfo/indimail-support. You can email indimail-support@lists.sourceforge.net for posting messages to this list. Old discussions can be seen here.
2. indimail-devel - You can subscribe at https://lists.sourceforge.net/lists/listinfo/indimail-devel. You can email indimail-devel@lists.sourceforge.net for posting messages to this list. Old discussions can be seen here.
3. Archive at Google - http://groups.google.com/group/indimail. This group acts as a remote archive for both indimail-support and indimail-devel lists at sourceforge.net. Any discussions posted here goes to indimail-support.
4. indimail-announce - This is only meant for announcement of New Releases or patches. You can subscribe at http://groups.google.com/group/indimail.
5.

Lists Archive

IndiMail has two List Archives which you can browse at

indimail-suppport - https://sourceforge.net/mailarchive/forum.php?forum_name=indimail-support

indimail-devel - https://sourceforge.net/mailarchive/forum.php?forum_name=indimail-devel

There is also a Project Tracker for IndiMail (Bugs, Feature Requests, Patches, Support Requests) at
http://sourceforge.net/tracker/?group_id=230686

2. Installation

This section covers installing IndiMail. If you're an experienced system administrator, you can install IndiMail following the directions below. If you are in a hurry, you can directly jump to the Section 'Installation Steps' below. However, I would recommend you to go through the Checklist below too to understand what is involved in setting up a fully fledged mail server.

Note: If you choose to install using the following directions, you should read through the entire section to familiarize yourself with the overall process.

2.1 Checklist

You need to have answers to the following ready before starting the installation (I need to bring more clarity to this
section so that it becomes easy for a novice to install a mail server)

1. Whether you want a full fledged mail server or just a relay server installation
2. Default Domain name for your Mail Server (mailserver)
    1. postmaster email account
    2. abuse account

3. What Filesystem to use (XFS, EXT4 for Maildirs, ext2 for queue)? (mailserver)
4. Filesystem where you will have your queue (mailserver, relayserver)
5. Filesystem where you will have your user's home directory (mailserver)
6. Filesystem where you will have the MySQL data files and logs (mailserver, clusterinfo)
7. Filesystem where you will have the supervise log files
8. A MySQL database to hold the user cluster information (clusterinfo)
9. A MySQL database to hold local user information for each host which is part of the user cluster (mailserver)
10. Userids, passwords for the MySQL Database (mailserver, clusterinfo)
11. Estimation of Load (mailserver, relayserver, clusterinfo)
12. IP addresses (Mailserver, relayservers, clusterinfo)
13. MX records to be setup (relayserver)
14. Access to root (mailserver, relayserver, clusterinfo)
15. Broad level features required (SPAM, IMAP, POP3, WebMail) (mailserver)
16. Whom should the daily mail statistics reports be sent to.

TERMS

- mailserver - A host which keeps the user's maildir
- relayserver - A host which accepts mail from the internet (port 25) and/or the users (port 587)

- clusterinfo - A host which hosts the MySQL database having the user cluster information (only if you want to install a single domain multihost mailserver

Before starting the installation, there are a few things you need to think about.

- If possible, install IndiMail on a "practice" system. This will give you a chance to make mistakes without losing important mail or interrupting mail service to your users. You can also use DESTDIR=staging_directory to install indimail in a staring directory before copying to the actual destination.
- If you don't have a spare, and your system is already handling mail using sendmail, smail, or some other MTA, you can install and test most pieces of IndiMail without interfering with the existing service.
- When migrating a system from some other MTA to IndiMail--even if you've got some IndiMail experience under your belt--it's a good idea to formulate a plan.

2.2. System requirements

IndiMail will install and run on most UNIX and UNIX-like systems, but there are few requirements:

- Around 300 megabytes of diskspace for all the packages (you will download this in the build area).
- About 100 megabytes of free space in the build area during the build. After the build, you can free about 15 megabytes by doing make clean
- A complete, functioning C development system including a compiler, system header files, and libraries. The installation steps will guide you through the installation process.
- Around 80 megabytes for the binaries, documentation, and configuration files.
- A safe filesystem for the queue. qmail's reliability guarantee requires that the queue reside on a filesystem with traditional BSD FFS semantics. Most modern local filesystems meet these requirements with one important exception: the link() system call is often asynchronous--meaning that the results of the link() operation might not have been written to disk when the link() call returns. Bruce Guenter's syncdir library can be used to work around this problem.
- Sufficient disk space for the queue. Small single-user systems only need a couple megabytes. Large servers may need a couple gigabytes.
- A filesystem for the user's home directories where mail will be delivered.
- A compatible operating system. Most flavors of UNIX are acceptable which have GNU Compilation tools (autoconf/automake/libtool/texinfo/emacs). NOTE: autoconf 2.6 and above is required.
- Access to a domain name server (DNS) is highly recommended. Without one, qmail can only send to remote systems configured in its smtproutes config file.
- Adequate network connectivity. IndiMail was designed for well-connected systems, so you probably don't want to try to use it for a mailing list server on

a 28.8k dial-up. The fetchmail package was designed to make Mail more compatible with poorly-connected systems works well with IndiMail. The installation and configuration for fetchmail is also discussed. Unix Development Environment (C compiler, make, etc) and other commands like gzip, bzip2, tar, vi, etc. If you can't find a compiler installed, you'll have to locate one and install it. Contact your administrator or OS vendor.

- IndiMail supports the concept of staged installation. You need to specify make DESTDIR=path_to_staging_directory when doing the build. Also svctool (see below) can be passed an extra argument --destdir=path_to_staging_directory to create/modify all configuration files in the staging area alone. Using staged installation, the administrator can install IndiMail on a live system without disturbing an existing running installation. To upgrade, all that is required is to move all files from the staged directories to the actual production directories. The Directory tree structure staged area is exactly the same as would be present in the production directory.

2.3 Installation Steps

2.3.1 Download INSTALL Instructions

It could be handy to download the instructions from IndiMail Google Groups and familiarize yourself with installation instructions. For further details, you may want to read section 2.4.2 below. The INSTALL file may unergo frequent updations. The latest will always be found at http://groups.google.com/group/indimail/files

2.3.2. Configuration Settings

The behaviour of IndiMail is determined by several configuration options set while configuring IndiMail before the build. These options are set using the configure program. i.e.

```
% configure [options]
% make
% make install (or make install-strip)
```

The current values used by the build are stored in the file indimail.settings. This file is created when you run the configure script.

The table below gives the values of the various configuration parameters used by IndiMail.

Configuration Name	***Option for setting this value***	***Default Value***	***Description***
qmail directory	--enable-qmaildir= homedir_of_qmail	--enable-qmaildir=/var/indimail	Directory where IndiMail will look for qmail binaries and configuration.
postfix directory	--enable-postfixdir	--enable-postfixdir=/var/postfix	Directory where IndiMail will look for postfix executables if MTA used is postfix.
IndiMail user	--enable-indiuser=user	--enable-indiuser=indimail	User who can run IndiMail binaries
IndiMail group	--enable-indigroup=group	--enable-indigroup=vchkpw	Group of the user who can run IndiMail binaries
default domain	--enable-default-domain= domain_name	no defaults	Sets the default domain when domain is not specified in email addresses.
syslog log name	--enable-log-name=user	--enable-log-name=indimail	Sets the default name used by the syslog function
mysqlprefix	--enable-mysqlprefix=/usr/local/mysql	--enable-mysqlprefix=/usr/local/mysql	Sets the MySQL Prefix directory
include dir	--enable-mysqlincdir=include_path	--enable-mysqlincdir= /usr/local/mysql/include/mysql	Sets the -I flag for compiling IndiMail.
lib dir	--enable-mysqllibdir=lib_path	--enable-libdir= /usr/local/mysql/lib/mysql	Sets the -L and -R flag while linking IndiMail.
roaming users	--enable-roaming-users=y	--enable-roaming-users=y	If set, after successful authentication, entry is made in the table ***relay***.
relay clear mins	--enable-relay-clear-minutes=mins	--enable-relay-clear-minutes=-60	**clearopensmtp** uses this value to clear entries in relay table older than relay-clear-minutes
tcprules program	--enable-tcprules-prog=path_of_tcprules	Searches for tcprules in /usr/local/bin, /usr/bin, /var/indimail/bin and uses the first found directory	program to build the cdb file
tcpserver file	--enable-tcpserver-file= path_of_cdb_file	Searches for tcp.smtp in ~indimail/etc, /etc, /etc/tcprules.d and uses the first found directory.	Access control file used by tcpserver.
open smtp file	cannot be altered during compilation.	~indimail/etc/open-smtp	Temporary file used by IndiMail to build the cdb file.

Configuration Name	***Option for setting this value***	***Default Value***	***Description***
pop syslog	--enable-logging=y|e|p|n	--enable-logging=e	Sets the logging level y – show successful and failure login attemts e – show only failure logging attemts p – log everything including passwords in failures n – logging off
auth logging	--enable-auth-logging=y	--enable-auth-logging=y	If set, all successful authentication is logged to ***lastauth*** table.
mysql logging	--enable-mysql-logging=y	--enable-mysql-logging=n	Causes vlog() function to log to mysql in addition to syslog
user quota	--enable-hardquota= quota_value	No default	Enables the quota mechanism in IndiMail. Also the default quota is set to 'n' for users for whom quota is not given during creation.
maildir++ quota	--enable-maildir-quota=y	--enable-maildir-quota=y	uses the maildir quota calculation used by courier-imap. Increases the efficiency of mail deliveries
large site	--enable-large-site=y	--enable-large-site=n	causes separate tables for each domain. Use this if you have more than 10000 domains and users more than 1 Million in each domain.
clustered architechture	--enable-user-cluster=y	--enable-user-cluster=n	If enabled, domains can be clustered across multiple machines.
address extensions	--enable-qmail-ext=y	--enable-qmail-ext=y	Enables qmail address extensions
ip alias	--enable-ip-alias-domains=y	--enable-ip-alias-domains=n	
valias processing	--enable-valias=y	--enable-valias=y	Enables valias (MySQL version of .qmail files).
vfilter processing	--enable-vfiler=y	--enable-vfilter=n	Enables IndiMail Filter mechanism during delivery.
file locking	--enable-file-locking=c	--enable-file-locking=c	Enables locking during update of configuration files and use fcntl() for file locking.
file sync	--enable-file-sync=y	--enable-file-sync=y	Syncs the file after mail delivery to guarantee delivery.

Configuration Name	*Option for setting this value*	*Default Value*	*Description*
make input seekable	--enable-make-seekable=y	--enable-make-seekable=y	Causes the input to vdelivermail to be made seekable even if the STDIN is not seekable.
query balancing	--enable-random-balancing=y	--enable-random-balancing=n	Causes queries to InLookup to be balanced randomly across multiple InLookup processes. If not set, queries time sliced balancing is done.
query cache	--enable-query-cache=y	--enable-query-cache=n	Causes inquery, auth modules to cache passwd queries across multiple requests for a user. Not desirable if multple InLookup is used and user changes password.
system passwords	--enable-passwd=y	--enable-passwd=y	Enables system users for Authenticated SMTP.
domain limits	--enable-domain-limits=y	--enable-domain-limits=y	Sets limits for administration of domains
password hash	--enable-password-hash=des| md5|sha256|sha512	--enable-password-hash=md5	Sets the hash method for encryption
Mysql Escape	--enable-mysql-escape=y| n	--enable-mysql-escape=y	Use mysql_real_escape_string () API to prevent SQL injection vulnerability
MD5 crypt	--enable-md5-crypt=y | n	--enable-md5-crypt=n	enable internal md5_crypt() function
SHA256	--enable-sha256-crypt=y | n	--enable-sha256-crypt=n	enable internal sha256_crypt() function.
SHA512	--enable-sha512-crypt=y | n	--enable-sha512-crypt=n	enable internal sha512_crypt() function.
IPV6	--enable-ipv6=y|n	--enable-ipv6=y	Enables IPV6 code
LOGDIR	--enable-logdir=dir	--enable-logdir=/var/log/indimail	Sets the directory where all logs for supervised services get created
BASEPATH	--enable-basepath=dir	--enable-basepath=/home/mail	Default directory for user's home directories

You are now ready to add virtual domains and virtual users and be able to send/receive mails for these users. But before sending/receiving mails you need an MTA.

2.3.3. Follow INSTALL instructions

If you follow the instructions in the INSTALL file downloaded in section 2.4.1, you should have a functional installation of IndiMail. You might want to change few settings to suit your requirements/needs.

2.3.4. Delivery Mode

The INSTALL instructions above creates default delivery to Maildir. If this is not what you want, at this point you need to decide the default delivery mode for messages that aren't delivered by a .qmail file. The following table outlines some common choices.

Mailbox Format	*Name*	*Location*	*defaultdelivery*	*Comments*
mbox	Mailbox	$HOME	./Mailbox	most common, works with most MUA's
maildir	Maildir	$HOME	./Maildir/	more reliable, less MUA support
mbox	username	/var/spool/mail	See INSTALL.vsm	traditional mailbox

See INSTALL.mbox, INSTALL.maildir, and INSTALL.vsm for more information.

To select your default mailbox type, just enter the defaultdelivery value from the table into /var/indimail/control/defaultdelivery. e.g., to select the standard Maildir delivery, do:

```
echo ./Maildir/ > /var/indimail/control/defaultdelivery
```

Note: defaultdelivery isn't a standard qmail control file. It's a feature startup scripts created by **svctool**. The defaultdelivery argument to qmail-start is the contents of a .qmail file that specifies delivery instructions to be followed when no actual .qmail is found. Putting these instructions in a separate control file eliminates the need to quote shell meta characters in the delivery instructions and avoids messy multi-line command arguments.

The last step is to create a couple of system aliases.

2.3.5. Create System Aliases

The instructions in INSTALL above would have created the default aliases using postmaster@indimail.org as the default mailbox. There are three system aliases that should be created on all IndiMail installations:

Alias	*Purpose*
postmaster	RFC821 required, points to the mail administrator (you)
mailer-daemon	de facto standard recipient for some bounces
root	redirects mail from privileged accounts to the system administrator

To create these aliases, decide where you want each of them to go (a local user or a remote address) and create and populate the appropriate .qmail files. For example, say local user mbhangui is both the system and mail administrator:

```
echo postmaster@indimail.org > /var/indimail/alias/.qmail-root
echo postmaster@indimail.org > /var/indimail/alias/.qmail-postmaster
ln -s .qmail-postmaster /var/indimail/alias/.qmail-mailer-daemon
chmod 644 /var/indimail/alias/.qmail-root /var/indimail/alias/.qmail-postmaster
```

See INSTALL.alias for more details.

2.3.6. Crontab entries

Following are the most common entries to be entered in cron.

If IndiMail has been configured with -enable-roaming-users

```
3,33 * * * * /var/indimail/bin/clearopensmtp > /dev/null 2>&1
```

The above entry helps in periodic cleanup of old entries in the relay table. The ***relay*** table has timestamps and IP addresses inserted for users who authenticate using POP3 or IMAP. These entries help in doing POP/IMAP before SMTP and help in relay control. Not enabling this entry could cause the relay table to build up resultig in performance degradation of SMTP.

If IndiMail has been configured with --enable-user-cluster

```
0,15,30,45 * * * * /var/indimail/bin/hostsync -d test.com > /dev/null 2>&1
```

The above entry ensures that users added/deleted and modified on the local host are synchronized on the ***hostnctrl*** MySQL table**.** Synchronization is required for the case where the host having ***hostcntrl*** is down for some reason when users are being added/deleted. This entry is needed only for clustered domains, one line for each clustered domain (added through **vadddomain**).

```
0 2 * * * /var/indimail/bin/svc -a /service*/*/log > /dev/null 2>&1
```

The above entry is required to rotate the logs created by IndiMail. You may increase or decrease the frequency depending on the volume of mails handled on your site and your requirement of log retention (if any).

```
0 0 * * * /var/indimail/bin/vdeloldusers -d test.com -u -1 -t 2 > /dev/null 2>&1
```

This entry is needed if you desire old users to be stopped from receiving mails on 30 days of inactivity (see **vdeloldusers** for more details). test.com has been given just as an example. Replace test.com with your own virtual domain created.

2.3.7. Start IndiMail

IndiMail uses supervise to startup all daemons. A script called supervise ensures that all daemons run continuously. The daemons get restarted automatically, in the case they go down due to some reason. A script called svscan needs to be run to start up supervise. If you have followed the INSTALL instructions above, you would have installed all necessary startup scripts to have IndiMail run in supervise mode. All you need now is to have IndiMail started by by init – by executing the **initsvc** command. **initsvc** also works for new OS where **init** has got replaced by **upstart**.

```
/var/indimail/bin/initsvc -on
```

NOTE: Passing the argument -off to initsvc, removes svscan from being executed by init.

You can run the following command for status

```
/var/indimail/bin/initsvc -status
```

In case you are not using daemontools, use the following command

```
/var/indimail/bin/qmailctl start
```

2.3.8. Test the Installation

IndiMail should now be running. First run qmailctl stat to verify that the services are up and running:

```
# /var/indimail/bin/qmailctl stat
        /service/clamd: up (pid 1844) 22230 seconds
        /service/fetchmail: up (pid 1823) 22230 seconds
        /service/freshclam: up (pid 1822) 22230 seconds
        /service/indisrvr.4000: up (pid 1832) 22230 seconds
        /service/inlookup.infifo: up (pid 1834) 22230 seconds
        /service/mysql.3306: up (pid 1813) 22230 seconds
        /service/proxy-imapd.4143: up (pid 1819) 22230 seconds
        /service/proxy-pop3d.4110: up (pid 1826) 22230 seconds
        /service/qmail-imapd.143: up (pid 1833) 22230 seconds
        /service/qmail-pop3d.110: up (pid 1814) 22230 seconds
        /service/qmail-send.25: up (pid 1838) 22230 seconds
        /service/qmail-smtpd.25: up (pid 1839) 22230 seconds
        /service/qmail-smtpd.366: up (pid 1815) 22230 seconds
        /service/qmail-spamlog: up (pid 1837) 22230 seconds
        /service/qscanq: up (pid 24482) 30 seconds
        /service/clamd/log: up (pid 1841) 22230 seconds
        /service/fetchmail/log: up (pid 1812) 22230 seconds
```

/service/freshclam/log: up (pid 1816) 22230 seconds
/service/indisrvr.4000/log: up (pid 1835) 22230 seconds
/service/inlookup.infifo/log: up (pid 1829) 22230 seconds
/service/mysql.3306/log: up (pid 1820) 22230 seconds
/service/proxy-imapd.4143/log: up (pid 1817) 22230 seconds
/service/proxy-pop3d.4110/log: up (pid 1827) 22230 seconds
/service/qmail-imapd.143/log: up (pid 1818) 22230 seconds
/service/qmail-pop3d.110/log: up (pid 1831) 22230 seconds
/service/qmail-send.25/log: up (pid 1836) 22230 seconds
/service/qmail-smtpd.25/log: up (pid 1824) 22230 seconds
/service/qmail-smtpd.366/log: up (pid 1811) 22230 seconds
/service/qmail-spamlog/log: up (pid 1821) 22230 seconds
/service/qscanq/log: up (pid 1825) 22230 seconds

All above services should be "up" for more than a second. If they're not, you've probably got a typo in the associated run script or you skipped one or more steps in creating the necessary files, directories, or links. Go back through the installation step-by-step and double check your work.

The readproctitle program keeps a log of error messages generated by services managed by svscan. To see these messages, use ps or some other process listing command. For example, you might see something like:

```
# ps -efl | grep "service errors" | grep -v grep
root     1775     1  0 08:28 ?        00:00:00 /var/indimail/bin/readproctitle /service
errors: ..........................................................................................................
.....................................................................................................
.....
```

It sometimes helps to run a service manually in order to find configuration problems. For example, if your qmail-smtpd/log service isn't running, do:

```
cd /service/qmail-smtpd.25/log
/var/indimail/bin/svc -d .
./run
if no errors, enter a line of text and press ENTER
if still no errors, enter CTRL-D (end of file)
```

At this point, you should be able to identify the problem and fix it. Once that's done, return to the service's directory, if necessary, and do:

```
/var/indimail/bin/svc -u .
```

Next, follow the instructions in TEST.deliver and TEST.receive to verify that it's working correctly. Note that using these instructions, logging will be accomplished by **multilog.**

Note: If you chose maildir mailbox format as the default delivery method, you will need to create a Maildir directory in your home directory and alias's home directory before trying these instructions. See the maildir section to see how to properly create this directory.

3. Configuration

This section contains information you will need to configure IndiMail to make it work the way you want it to.

3.1. Control Files

All of IndiMail's system configuration files, with the exception of the .qmail files in ~alias, reside in /var/indimail/control. This location can be changed by specifying the environment variable CONTROLDIR. The qmail-control man page contains a table like the following:

Control	***Default***	***Used By***	***Purpose***
badmailfrom	none	qmail-smtpd	blacklisted "From" addresses
badmailpatterns	none	qmail-smtpd	Same as above except that wildcard patterns can be given
blackholedsender	none	qmail-smtpd, vdelivermail	Blacklisted "From" addresses. Additionally, the SMTP session is terminated without acknowledgement to client.
blackholedpatterns	none	qmail-smtpd, vdelivermail.	Same as above except that wildcard patterns can be given
badheaders	none	qmail-smtpd	Spam ranking for various headers. Mail is rejected if sum total of all rankings exceed value of badheaderthresh
badheaderthresh	none	qmail-smtpd	Maximum spam ranking allowed
queueforward	none	qmail-smtpd	Patterns which if mathes with "From" address, mail is forwarded to a different queue as specified in queueforward.
badrcptto	none	qmail-smtpd	blacklisted "To" addresses
badrcptpatterns	none	qmail-smtpd	Same as above except that wildcard patterns can be given

Control	*Default*	*Used By*	*Purpose*
bouncefrom	MAILER-DAEMON	qmail-send	username of bounce sender
bouncehost	me	qmail-send	hostname of bounce sender
bouncesubject	"failure notice"	qmail-send	subject for failure notices
bouncemessage	QSBMF	qmail-send	the bouncemessage text
doublebouncesubject	"failure notice"	qmail-send	subject for double bounce
doublebouncemessage	QSBMF	qmail-send	text for double bounce
bouncemaxbytes	none	qmail-send	If bounce text is greater than this size, the bounce is truncated
concurrencylocal	10	qmail-send	max simultaneous local deliveries
concurrencyremote	20	qmail-send	max simultaneous remote deliveries
concurrencyincoming	none	qmail-smtp supervise run script	max simultaneous incoming SMTP connections.
defaultdelivery	none	qmail-send supervise run script	default .qmail file
defaultdomain	me	qmail-inject	default domain name
defaulthost	me	qmail-inject	default host name
databytes	0	qmail-smtpd, qmail-inject, sendmail	Max number of bytes in message (0=no limit)
maxhops	100	qmail-smtpd	Messages with hops greater than this number is rejected
doublebouncehost	me	qmail-send	host name of double bouncesender
doublebounceto	postmaster	qmail-send	user to receivedouble bounces
envnoathost	me	qmail-send	default domain for addresses without "@"
helohost	me	qmail-remote	host name used in SMTP HELO command
idhost	me	qmail-inject	host name for Message-ID's
localiphost	me	qmail-smtpd	name substituted for local IP address

Control	*Default*	*Used By*	*Purpose*
moreipme	none	qmail-smtpd	addresses to be treated as a local IP address.
notipme	none	qmail-smtpd	address not to be treated as a local IP address. This has precedence over moreipme.
outgoingip	me	qmail-remote	Outgoing IP to use for delivering remote mails. Used with hosts having multiple ethernet interfaces
bindroutes	none	qmail-remote	Outgoing ip to use for certain remote addresses to which mails are to be delivered.
domainbindings	none	qmail-remote	Outgoing ip to use for certain remote domains to which mails are to be delivered.
locals	me	qmail-send	domains that we deliver locally
me	FQDN of system	various	default for many control files
morercpthosts	none	qmail-smtpd	secondary rcpthosts database
percenthack	none	qmail-send	domains that can use "%"-style relaying
nodnscheck	none	qmail-smtpd	Domains excluded from domain name validification
plusdomain	me	qmail-inject	domain substituted for trailing "+"
qmqpservers	none	qmail-qmqpc	IP addresses of QMQP servers
queuelifetime	604800	qmail-send	seconds a message can remain in queue
rcpthosts	none	qmail-smtpd	domains that we accept mail for
smtpgreeting	me	qmail-smtpd	SMTP greeting message
smtproutes	none	qmail-remote	artificial SMTP routes
timeoutconnect	60	qmail-remote	how long, in seconds, to wait for SMTP connection

Control	*Default*	*Used By*	*Purpose*
timeoutremote	1200	qmail-remote	how long, in seconds, to wait for remote server
timeoutsmtpd	1200	qmail-smtpd	how long, in seconds, to wait for SMTP client
timeoutread	4	qmail-smtpd	How long, in seconds, to wait for query from InLookup
timeoutwrite	4	inlookup	How long will inquery wait for writing to client fifo
virtualdomains	none	qmail-send	virtual domains and users
etrnhosts	none	qmail-smtpd	domains for which ETRN/ODMR is enabled.
chkrcptdomain	none	qmail-smtpd	domains for which 'User Status' checking is enabled.
relayclients	none	qmail-smtpd	Hosts allowed to relay
relaydomains	none	qmail-smtpd	Domains allowed to relay
relaymailfrom	none	qmail-smtpd	Relaying allowed for mails having this envelope sender
tarpitcount	none	qmail-smtpd	No of RCPT Tos to accept before starting tarpitting.
tarpitdelay	5	qmail-smtpd	No of seconds of delay to introduce after each subsequent RCPT TO once tarpitcount has been reached
extraqueue	none	qmail-queue	Content of this file is added to each recipient list.
holdremote	0	qmail-send	If set to 1, remote deliveries are held
holdlocal	0	qmail-send	If set to 1, local deliveries are held
badhelo	none	qmail-smtpd	HELO/EHLO strings matching REGEX patterns in this file are rejected

Control	*Default*	*Used By*	*Purpose*
signaturedomains	none	qmail-dkim	List of domains which are known to have DKIM-Signature published
nosignaturedomains	none	qmail-dkim	List of domains which have not implemented dkim.
etrnhosts	none	qmail-smtpd	
badext	none	qsanq-stdin	
badextpatterns	none	qscanq-stdin	
maxrecipients	none	qmail-smtpd	
greetdelay	none	qmail-smtpd	
qregex	none	qmail-smtpd spawn-filter	
signatures	none	qmail-smtpd	
bodycheck	none	qmail-smtpd	
hostaccess	none	qmail-smtpd	
spamfilter	none	mail-multi spawn-filter	
spamignore	none	qmail-multi spawn-filter	
spamignorepatterns	none	qmail-multi spawn-filter	
spamredirect	none	qmail-multi spawn-filter	
rejectspam	none	qmail-multi spawn-filter	
accesslist	none	qmail-smtpd	
authdomains	none	qmail-smtpd	
goodrcpt	none	qmail-smtpd	
goodrcptpatterns	none	qmail-smtpd	
blackholedrcpt	none	qmail-smtpd	
blackholedrcptpatterns	none	qmail-smtpd	
quarantine	none	qmail-queue	
chkrcptdomains	none	qmail-smtpd	
signkey	none	qmail-remote qmail-smtpd	
signkeystale	7	qmail-smtpd	
nosignhosts	none	qmail-remote qmail-smtpd	
nosignmydoms	none	qmail-remote	

Control	*Default*	*Used By*	*Purpose*
greylist.white	none	qmail-greyd	List of whitelisted IP addresses

For more information about a particular control file, see the man page for the module listed under "Used by".

3.2. Environment Variables

Some IndiMail programs set or use environment variables. The following table lists these variables and describes their use.

Table 1Environment Variables and Programs using them

	Name	***Program***	***Set/Used***	***Purpose***
1.	DATABYTES	qmail-smtpd qmail-inject sendmail spawn-filter	used	overrides control/databytes
2.	DEFAULT	qmail-command	set	Portion of address matching "-default" in a .qmail file name.
3.	DTLINE	qmail-command	set	Delivered-To header field
4.	EXT	qmail-command	set	The address extension
5.	EXT2	qmail-command	set	Portion of EXT following first dash
6.	EXT3	qmail-command	set	Portion of EXT following second dash
7.	EXT4	qmail-command	set	Portion of EXT following third dash
8.	HOME	qmail-command	set	The user's home directory
9.	HOST	qmail-command	set	The domain part of the recipient address
10.	HOST2	qmail-command	set	Portion of HOST preceding last dot.
11.	HOST3	qmail-command	set	Portion of HOST preceding second-to-last dot
12.	HOST4	qmail-command	set	Portion of HOST preceding third-to-last dot
13.	LOCAL	qmail-command	set	The local part of the recipient address
14.	NEWSENDER	qmail-command	set	Forwarding sender address (see "man dot-qmail")
15.	RECIPIENT	qmail-command	set	Envelope recipient address
16.	RPLINE	qmail-command	set	Return-Path header field
17.	SENDER	qmail-command	set	Envelope sender address
18.	UFLINE	qmail-command	set	UUCP-style "From" line
19.	USER	qmail-command	set	The current user
20.	LOGNAME	qmail-inject	used	user name in From header field (4)
21.	MAILHOST	qmail-inject	used	Host name in From header field (2)
22.	MAILNAME	qmail-inject	used	Personal name in From header field (2)

	Name	*Program*	*Set/Used*	*Purpose*
23.	MAILUSER	qmail-inject	used	User name in From header field (2)
24.	NAME	qmail-inject	used	Personal name in From header field (3)
25.	QMAILDEFAULTDOMAIN	qmail-inject	used	Overrides control/defaultdomain
26.	QMAILDEFAULTHOST	qmail-inject	used	Overrides control/defaulthost
27.	QMAILHOST	qmail-inject	used	Host name in From header field (1)
28.	QMAILIDHOST	qmail-inject	used	Overrides control/idhost
29.	QMAILINJECT	qmail-inject	used	Specify various options (see next table)
30.	QMAILMFTFILE	qmail-inject	used	File containing list of mailing list addresses for Mail-Followup-To generation
31.	QMAILNAME	qmail-inject	used	Personal name in From header field (1)
32.	QMAILPLUSDOMAIN	qmail-inject	used	Overrides control/plusdomain
33.	QMAILSHOST	qmail-inject	used	Host name in envelope sender address
34.	QMAILSUSER	qmail-inject	used	User name in envelope sender address
35.	QMAILUSER	qmail-inject	used	User name in From header field (1)
36.	USER	qmail-inject	used	User name in From header field (3)
37.	EXTRAQUEUE	qmail-queue	used	Extra recipient to be added to the recipient list before queueing the mail.
38.	QHPSI	qmail-queue	used	If set enables the Qmail High Performance Scanner Interface. This should be set to the path of the virus scanner. If SCANCMD is defined
39.	QHPSIRC	qmail-queue	used	To specify the return code of the virus scanner in case of an infection; default is 1.
40.	QHPSIMINSIZE	qmail-queue	used	The minimal size of the message to invoke the virus scanner; default is 0. A typical choice would be QHPSIMISIZE=10000 (~10k).

	Name	Program	Set/Used	Purpose
41.	QHPSIMAXSIZE	qmail-queue	used	The maximal size of the message to invoke the virus scanner; default is unrestricted. A typical choice would be QHPPIMAXSIZE=1000000 (~1M).
42.	QUARANTINE	qmail-queue	used	If set, all mails get redirected to the id defined by the environment variable
43.	QHPSIFORWARD	qmail-queue	used	If set, virus infected mails get redirected to the id defined by the environment variable
44.	REJECTVIRUS	qmail-queue	used	Sets action to be carried out on detection of virus 0 – Accept infected mails 1 – Bounce the mail. 2 – Blackhole the mail.
45.	NULLQUEUE	qmail-queue	used	If set, mail transaction is sent to /dev/null without returning an error (blackholing)
46.	ENVHEADERS	qmail-queue		
47.	LOGHEADERS	qmail-queue		
48.	REMOVEHEADERS	qmail-queue	used	Name of control file containing a list of email headers that should be removed before putting into the queue. If not supplied, qmail-queue uses the control file removeheaders.
49.	PLUGINDIR	qmail-queue	used	Directory in /var/indimail containing dynamically loadable virus scanner plugins
50.	VIRUS_PLUGIN	qmail-queue	used	The function to be called when a dynamic virus scanner plugin is loaded
51.	QQEH	qmail-queue qmail-local forward condredirect	used	qmail-queue passes this variable into the queue. qmail-local and qmail-remote will prepend it to the headers of email when it is delivered

	Name	*Program*	*Set/Used*	*Purpose*
52.	QUEUEDIR	qmail-queue qmail-clean qmail-daemon qmail-rm qmail-send qmail-tcpok qmail-tcpto qmail-todo spawn-filter qmail-lspawn qmail-rspawn	used	queue path from where to pick up mails for local and remote deliveries
53.	DKQUEUE	qmail-dk	used	Specifies the path to the **qmail-queue** executable. If this is not set, **qmail-dk** will invoke **qmail-multi**
54.	DKSIGN	qmail-dk	used	If set to the path of a private key, **qmail-dk** signs the message using the private key. This variable should be set when a host is authorized to relay.
55.	SIGN_PRACTICE	qmail-dkim	used	For using SSP or ADSP as the signing practice
56.	DKVERIFY	qmail-dk	used	If set, messages are verified and **qmail-dk** will insert the **DomainKey-Status** header. The behaviour depends on the set of letters to which **DKVERIFY** is set. A conservative set of letters is **DEGIJKfh**
57.	DKSIGNOPTIONS	qmail-dk dk-filter	used	Options to use for dk signing.
58.	DKIMQUEUE	qmail-dkim	used	Specifies the path to the **qmail-queue** executable. If this is not set, **qmail-dkim** will invoke **qmail-multi**
59.	DKIMSIGN	qmail-dkim	used	If set to the path of a private key, **qmail-dkim** signs the message using the private key. This variable should be set when a host is authorized to relay.
60.	DKIMVERIFY	qmail-dkim	used	If set, messages are verified and **qmail-dkim** will insert the **DKIM-Status** header.
61.	DKIMSIGNOPTIONS	qmail-dkim dk-filter	used	Options to use for dkim signing
62.	QUEUE_COUNT	qmail-multi	used	No of queues setup

	Name	Program	Set/Used	Purpose
63.	QUEUE_START	qmail-multi	used	Number denoting the first queue
64.	QUEUEDIR	qmail-multi	set	queue path where mails are queued
65.	MIN_FREE	qmail-multi	used	Minimum free disk space allowed on the QUEUE filesystem after which mails get rejected with a temporary error. By setting this the IndiMail system becomes crashproof as well as bounceproof
66.	SPAMFILTER	qmail-multi spawn-filter	used	Filter program to filter mail before passing it to qmail-queue
67.	SPAMFILTERARGS	qmail-multi spawn-filter	used	Arguments to be passed to the spam filter program.
68.	SPAMEXITCODE	qmail-multi spawn-filter	used	Exit code of the spamfilter which denotes a spam mail.
69.	REJECTSPAM	qmail-multi spawn-filter	used	If set, and if the filter program returns an exit code matching the value of SPAMEXITCODE, the mail is permanently rejected
70.	SPAMIGNORE	qmail-smtpd spawn-filter chowkidar	used	ignore users listed in this control file
71.	SPAMIGNOREPATTERNS	qmail-smtpd spawn-filter chowkidar	used	same as above, except that the user list can have wildcards.
72.	SPAMREDIRECT	qmail-multi spawn-filter	used	Redirect SPAM tagged mails to the user specified by this environment variable
73.	FILTERARGS	qmail-multi spawn-filter	used	Arguments with with Filter program will be called
74.	LOGFILTER	qmail-smtpd spawn-filter	used	If set, the filter program has available the file descriptor 255 where it can log additional status messages to be logged in the qmail-smtpd's log file. This should be set to a filename which will be used byu the spamlogger service.
75.	MAKE_SEEKABLE	qmail-multi spawn-filter vdelivermail qmail-autoresponder	used	If set, the pipe used by various programs to communicate is made seekable. This improves the performance of filters.

	Name	Program	Set/Used	Purpose
76.	QMAILLOCAL	spawn-filter	used	The mda to call for local deliveries (default qmail-local if not defined)
77.	QMAILREMOTE	spawn-filter	used	The smtp cilent to call for local deliveries (default qmail-remote if not defined)
78.	TMPDIR	qmail-multi qmail-qfilter qmail-smtpd spawn-filter	used	Sets directory to use for temporary files. Default is /tmp
79.	PROTO	sendmail	used	Used internally by sendmail -bs flag.
80.	CONTROLDIR	qmail-smtpd qmail-remote spawn-filter qmail-local ipmeprint qmail-newmrh qmail-qstat qmail-send qmail-showctl qmail-spamdb qmail-todo qmail-command	used	The directory containing control files. If not defined qmail_home/control is used
81.	RELAYCLIENT	qmail-smtpd	used	Ignore control/rcpthosts and append value to recipient address. This is used for selective relaying to domains not listed in rcpthosts.
82.	AUTH_ALL	qmail-smtpd	used	If set, sending mail to all domains will require authentication (regardless of whether the domain is listed in rcpthosts).
83.	REQUIREAUTH	qmail-smtpd	used	If set, 'mail from' is not accepted without prior authenticated SMTP.
84.	AUTH_DOMAINS	qmail-smtpd	used	List of comma separated domains whose senders will allways require authentication to send out mails (regardless of whether the domain is listed in rcpthosts). Multiple domains can also be set with ':' or newline as the delimiter. e.g. AUTH_DOMAINS=`cat /var/indimail/control/authdomains` where authdomains is a file containing list of domains one per line.

	Name	*Program*	*Set/Used*	*Purpose*
85.	CHECKRELAY	qmail-smtpd	used	Relay table in MySQL is checked for 'return path' before sending mails to outside domains. Allows users to relay mails to foreign domains.
86.	CHECKRECIPIENT	qmail-smtpd	used	Used for checking the status of the user. The presence of the user can be checked in MySQL (***indimail***) or */var/indimail/users/recipients.cdb*. If the control file ***chkrcptdomains*** is present, then only domains listed in chkrcptdomain are checked. If chkrcptdomain is absent, all domains are checked. The value of CHECKRECIPIENT can be one of the 3 values 1. Check MySQL 2. Check both MySQL and recipients.cdb 3. Check recipients.cdb
87.	MASQUERADE	qmail-smtpd	used	Allows the authenticated userid in AUTH SMTP to be different from the Return-Path.
88.	CUGMAIL	qmail-smtpd	used	Turns of relaying to foreign domains.
89.	ANTISPOOFING	qmail-smtpd	used	Turns of spoofing of Return-Path for local users.
90.	MAXRECIPIENTS	qmail-smtpd	used	Maximum number of recipients allowed in a single SMTP session.
91.	TARPITCOUNT	qmail-smtpd	used	Maximum number of recipients allowed in a single SMTP session after which an artificial delay is introduced.
92.	TARPITDELAY	qmail-smtpd	used	No of seconds of delay to be imposed on the SMTP session when number of recipients in a single SMTP session exceeds the value of TARPITCOUNT
93.	BADHELOCHECK	qmail-smtpd	used	Enables checking of badhelo control file and validity of the HELO/EHLO string as a valid domain.
94.	BADHELO	qmail-smtpd	used	Use this as the filename for bad helo REGEX strings.

	Name	*Program*	*Set/Used*	*Purpose*
95.	ENFORCE_FQDN_HELO	qmail-smtpd	used	Forces argument to helo/ehlo command to be a fully qualified domain name.
96.	BOUNCEMAIL	qmail-smtpd	used	When this is set, SMTP server issues this as a message to the client with a permanent error.
97.	BOUNCEPROCESSOR	qmail-send	used	External bounceprocess to call for handling bounces.
98.	BOUNCEQUEUE	qmail-send	used	Queue to be used for injecting bounces
99.	WARNMAIL	qmail-smtpd	used	Enables warn_mail function.
100.	WARNMAIL1 to WARNMAIL9	qmail-smtpd	used	X-Spam-Warning: lines inserted in message if these are set.
101.	SHUTDOWN	qmail-smtpd	used	Issues temporary errors for all commands
102.	BADMAILFROM	qmail-smtpd	used	Override the default name for the '***badmailfrom***' control file
103.	BADMAILPATTERNS	qmail-smtpd	used	Override the default name for the '***badmailpatterns***' control file
104.	BADRCPTTO	qmail-smtpd	used	Override the default name for the '***badrcptto***' control file
105.	BADRCPTPATTERNS	qmail-smtpd	used	Override the default name for the '***badrcptpatterns***' control file
106.	BODYCHECK	qmail-smtpd	used	Enables content filtering during a SMTP session. The value of BODYCHECK environment variable specifies the name of the control file containing regex patterns to be matched in an email.

	Name	*Program*	*Set/Used*	*Purpose*
107.	VIRUSCHECK	qmail-smtpd	used	Enables in-built virus scanning engine in qmail-smtpd. Virus patterns can be maintained in the control file '***signatures***'. If set to a value > 1, it can be used to call an external scan engine for checking viruses and bad attachments 1. Internal scanner 2. Internal+External+attachment scan. 3. Internal+attachment scan 4. External+attachment scan 5. External scanner 6. Attachment scan
108.	SIGNATURES	qmail-smtpd	used	Override the name of the default virus pattern control file '***signatures***'
109.	HOSTACCESS	qmail-smtpd	used	Defines from which IP address or set of IP addresses can mail from a particular domain originate.
110.	PARANOID	qmail-smtpd	used	If set, then only pairs having a match in the '***hostaccess***' control file are granted access
111.	DOMAIN_MASQUERADE	qmail-smtpd	used	If set, all IP address having a match in the '***hostaccess***' control file are granted access
112.	DOMAINBINDINGS	qmail-remote	used	Name to be used for the domainbindings control file.
113.	STARTTLS	qmail-smtpd	used	Enables the ESMTP STARTTLS extension
114.	SMTPS	qmail-smtpd	used	Turns on the deprecated SMTPS protocol on port 465
115.	TLSCIPHERS	qmail-smtpd	used	A set of OpenSSL cipher strings. Multiple ciphers contained in a string should be separated by a colon.
116.	NODNSCHECK	qmail-smtpd	used	If set, the domain part of an email address is not verified.

	Name	*Program*	*Set/Used*	*Purpose*
117.	SPFBEHAVIOUR	qmail-smtpd	used	SPF check gets enabled 0. SPF Disabled 1. Annotate with Received-SPF fields 2. Producte temporary failures on DNS lookup problems 3. Reject if SPF record says fail 4. Strict mode. Reject for 'fail' and 'softfail' 5. Reject for 'fail', 'softfail' and 'neutral' 6. Reject if no SPF record is available or for syntax error
118.	OPENRELAY	qmail-smtpd	used	If set, qmail-smtpd rejects incoming connection with (553 code) the message “No mail accepted from an open relay”
119.	QREGEX	qmail-smtpd spawn-filter	used	If set, regex is used to evaluate patterns in 1. badhelo 2. blackholedsender 3. badmailfrom 4. badrcptto 5. spamignore 6. relaymailfrom 7. authdomains 8. chkrcptdomains 9. accesslist

	Name	*Program*	*Set/Used*	*Purpose*
120.	ENVRULES	qmail-smtpd spawn-filter qmail-inject	used	Any environment variable(s) can be set based on senders/recipients matching a given value. SMTP REQUIREAUTH, QREGEX, ENFORCE_FQDN_HELO, DATABYTES, BADHELOCHECK, BADHELO, NODNSCHECK, VIRUSCHECK, SIGNATURES, BODYCHECK BADMAILFROM, BADMAILFROMPATTERNS, BOUNCEMAIL, CUGMAIL, MASQUERADE, DEFAULT_DOMAIN, BADRCPTTO, BADRCPTPATTERNS, TARPITCOUNT, TARPITDELAY, MAXRECIPIENTS, AUTH_ALL, CHECKRELAY, CONTROLDIR, ANTISPOOFING, CHECKRECIPIENT SPAMFILTER, LOGFILTER, SPAMFILTERARGS, SPAMEXITCODE, REJECTSPAM, SPAMREDIRECT, SPAMIGNORE, SPAMIGNOREPATTERNS, FILTERARGS, QMAILQUEUE, QMAILREMOTE, QMAILLOCAL, RELAYCLIENT, QQEH, BADEXT, BADEXTPATTERNS, ACCESSLIST, QHPSI, REJECTSPAM, REJECTVIRUS spawn-filter SPAMFILTER, LOGFILTER, SPAMFILTERARGS, FILTERARGS, SPAMEXITCODE, REJECTSPAM, SPAMREDIRECT, SPAMIGNORE, SPAMIGNOREPATTERNS, QREGEX, DATABYTES, SMTPROUTE
121.	SCANCMD	qscanq	used	The virus scanner to run for scanning emails. A '%s' in SCANCMD will get replaced with the filename in the mess directory (file containing the entire content of the email)
122.	BADEXT	qscanq-stdin	used	Name of control file specifying list of prohibited filename extensions in mail.
123.	BADEXTPATTERNS	qscanq-stdin	used	Name of control file specifying regex patterns of list of prohibited filename extensions in mail

	Name	*Program*	*Set/Used*	*Purpose*
124.	SCANDIR	qscanq	used	Name of directory where qscanq will create temporary work directory for extracting mime from emails.
125.	SCANINTERVAL	svscan	used	Time interval in which svscan should do a directory scan to look for new services. If set to 0, scanning is disabled. However the scan can be manually triggered by sending svscan a HUP signal. If not set, the default scan interval is 5 seconds.
126.	USE_FSYNC	qmail-queue qmail-local qmail-send qmail-todo	used	Causes fsync() to be used for all open files. This makes IndiMail crash proof with a minor performance penalty
127.	USE_SYNCDIR	qmail-queue qmail-local qmail-send qmail-todo	used	Emulates BSD style synchronous directories.
128.	TODO_INTERVAL	qmail-send qmail-todo	used	To set the minumum time interval between two todo runs. By increasing this you can reduce the disk IO. But there is a latency on mail delivery times
129.	ROUTES	qmail-rspawn	used	If defined as static, control/smtproutes is used, else hostcntrl is used and SMTPROUTES environment variables is set
130.	SMTPROUTE	qmail-rspawn	set	As above
131.	SMTPROUTE	qmail-remote	used	overrides control/smtproutes
132.	AUTH_SMTP	qmail-remote	used	Uses Authenticated SMTP on the remote SMTP server to push out mails. The username and password must be specified in smtproutes separated by spaces. e.g. test.com:25 postmaster@test.com pass
133.	AUTHSELF	qmail-lspawn	used	Causes PWSTRUCT environment variable to be set. This eliminates the need for vdelivermail to make connection to the database (MySQL). On a high volume server, this can significantly improve the database performance.

	Name	*Program*	*Set/Used*	*Purpose*
134.	PWSTRUCT	qmail-lspawn	set	Has /etc/passwd format. Set by qmail-lspawn after doing vauth_getpw() (fetch from MySQL).
135.	PWSTRUCT	vdelivermail vfilter	used	Uses the environment variable to fetch the passwd structure rather than from the database.
136.	MDA	vfilter	used	Executable to be called for delivering the mail. By default, **vfilter** calls **vdelivermail**. Command line arguments can also be specified in the environment variable
137.	MAILCOUNT_LIMIT	vdelivermail	used	Max deliveries permitted for a user in a day.
138.	MAILSIZE_LIMIT	vdelivermail	used	Max total size of mail delivery/day for a user.
139.	OVERQUOTA_MAILSIZE	vdelivermail	used	Maximum allowed size of mail when user is over quota
140.	HOLDOVERQUOTA	vdelivermail	used	If set, vdelivermail will defer mails when user runs out of quota. The command defined by OVERQUOTA_CMD will be run. However, BOUNCE_FLAG and lastdeliver table will not be updated
141.	MAILDIRFOLDER	vdelivermail	used	Alternate Folder to deliver incoming mail.
142.	MTA	vdelivermail	used	Default MTA to use for injecting mails (currently qmail-inject or postfix)
143.	OVERQUOTA_CMD	vdelivermail authindi	used	Command to be run when user becomes overquota. Triggered during mail delivery or during IMAP/POP3 authentication.
144.	REAL_DOMAINS	qmail-rspawn qmail-lspawn InLookup vfilter vdelivermail	used	List of real domains separated by the ':' character. e.g. indi.com:yahoo.com
145.	ALIAS_DOMAINS	qmail-rspawn qmail-lspawn InLookup vfilter vdelivermail	used	List of aliasdomain-realdomai pair separated by the ':' character. e.g. satyam.net.in,indi.com:yahoo.co.in,yahoo.com

	Name	*Program*	*Set/Used*	*Purpose*
146.	POSTAUTH	authindi systpass vchkpass proxyimap proxypop3	used	Command to be run post successful authentication. If return value is 2 or 3, the homedir gets set to the environment variable TMP_MAILDIR if defined else the user's home directoriy. If the return value is 3, authindi will additionally display a message defined by the environment variable MSG_ONERROR (if defined). Authindi will exit if the environment variable EXIT_ONERROR is defined.
147.	TMP_MAILDIR	authindi proxyimap proxypop3	used	Home directory set for imap/pop3 if the command defined by POSTAUTH exits with 2 or 3
148.	MSG_ONERROR	authindi proxyimap proxypop3	used	Message displayed to IMAP/POP3 clients when script defined by POSTAUTH exits with 2 or 3
149.	EXIT_ONERROR	authindi proxyimap proxypop3	used	If the command defined by POSTAUTH environment variable exits with 2 or 3 and this environment variable is defined, IMAP/POP3 client will exit preventing the user from accessing mailbox
150.	DESTPORT	authindi	used	Used to determine whether the destination port is imap or pop3. The format is either imap:port_num or pop3:port_num, where port_num is a port on which either IMAP or POP3 server is listening.
151.	NOLASTAUTH	authindi	used	If set, ***lastauth*** insert is disabled during authentication.
152.	MIN_LOGIN_TIME	authindi	used	If set to zero, users can authenticate via POP3/IMAP with any interval. If set to a value, minimum time of MIN_LOGIN_TIME has to elapse after which the user can authenticate.
153.	MIGRATEUSER	authindi	used	Program to be run once (first time) when the user logs in successfully (successful authentication).

	Name	Program	Set/Used	Purpose
154.	MIGRATEFLAG	authindi	used	Zero bytes file created in User's Maildir to figure out if the migration program defined by environment variable MIGRATEUSER has been run.
155.	ACTIVATEMAIL	authindi	used	Specifies the default name of the activation mail in the bulk mail directory.
156.	WELCOMEMAIL	authindi	used	Specifies the default name of the welcome mail in the bulk mail directory.
157.	BULK_MAILDIR	authindi	used	Constitutes part of the bulk mail directory. e.g. for indi.com the bulk mail directory will be /var/indimail/control/indi.com/$BULK_MAILDIR
158.	BULK_HOST BULK_VPORT BULK_SOCKET BULK_USER BULK_PASSWD BULK_DATABASE	authindi	used	Environment variables to be set when the '***bulkmail***' table lies on a remote MySQL host.
159.	MIN_LOGIN_INTERVAL	authindi	used	Minimum time permitted between two successful authentication. Authentication requests made before this time elapses are rejected. To remove this restriction, set this to 0.
160.	NORELAY	authindi	used	If set, insert to table '***relay***' after successful authentication is disabled. This in effect disables relaying (Roaming Users).
161.	TCP_FILE	authindi	used	File containing list of default tcprules to be applied. Default is /var/indimail/etc/tcp.smtp.
162.	OPEN_SMTP_CUR_FILE	authindi	used	File containing list of tcprules for IP addresses of users who have authenticated in the past RELAY_CLEAR_MINUTES. Default is /var/indimail/etc/open-smtp
163.	RELAY_TABLE	authindi InLookup clearopensmtp	used	MySQL table name containing user authentication timestamps for implementing POP/IMAP before SMTP.

	Name	Program	Set/Used	Purpose
164.	PASSWD_CACHE QUERY_CACHE	authindi InLookup vchkpass	used	Enables Password caching. Improves performance for a very busy site. For dormant sites, this should be turned off as passwd changes take time to get reflected.
165.	RELAY_CLEAR_MINUTES	clearopensmtp InLookup vuserinfo authindi vchkpass proxyimap proxypop3	used	No of minutes after which the authentication request maintained in the relay table is expired. This is used for sites using pop/imap before smtp authentication
166.	DATA_TIMEOUT	proxyimap proxypop3	used	Used for timing out if no data flows between the user's MUA and the proxy, or between the proxy and the actual IMAP/POP3 daemon.
167.	SLEEPTIME	proxyimap proxypop3	used	Minimum duration between connection attempts, if connection is refused.
168.	LEGACY_SERVER	proxyimap proxypop3	used	If the destination imap/pop3 server are not indimail then this needs to be set. The destination imap/pop3 server will always see proxy server's IP address regardless from where the user logs in.
169.	DOMAIN_LIMITS	authindi vadduser vdeldomain	used	The vlimit structure is checked and the pw_gid field is OR'ed with the value fetched from the vlimit table. This allows domain wide limits to be set for users.
170.	IMAPCLIENT	proxyimap	used	If set, access to IMAP is granted
171.	POP3CLIENT	proxypop3	used	If set, access to POP3 is granted
172.	ADMIN_HOST ADMIN_PORT ADMIN_USER ADMIN_PASS	proxyimap proxypop3	used	used by proxyimap and proxypop3 to connect to indisrvr for adding users. Also used by adminclient for executing administration commands.
173.	ADMIN_TIMEOUT	adminclient	used	Used to set timeout for executing any administration command by indisrvr
174.	AVG_USER_QUOTA	vfstab	used	value in bytes the average mailbox size for a user.

	Name	Program	Set/Used	Purpose
175.	HARD_QUOTA	proxyimap proxypop3 vadduser	used	Default value of quota for user addition through vadduser (without -q option).
176.	ALLOWCHARS	vadduser	used	List of allowed characters in the username.
177.	CNTRL_HOST CNTRL_USER CNTRL_PASSWD CNTRL_VPORT CNTRL_SOCKET CNTRL_TABLE CNTRL_DATABASE MASTER_HOST	InLookup vuserinfo qmail-rspawn qmail-lspawn		Environment variables to be set when a domain is distributed. These variables point to the controller host which has the '***hostcntrl***' table. If both MASTER_HOST and CNTRL_HOST are defined all updates go to MASTER_HOST and all queries go to CNTRL_HOST
178.	INFIFO	inquerytest InLookup qmail-smtpd authindi proxyimap proxypop3	used	Name of FIFO used for communicating with InLookup. InLookup pools connections to MySQL and provides the interface between IndiMail and MySQL
179.	DEBUG	InLookup vckpass	used	Enables verbose debugging
180.	EDITOR	dbinfo	used	Editor to use for editing the MCD file
181.	HOME	mate osh supernotepad secpanel lite	used	These programs uses the HOME environment variable to switch its CWD.
182.	MAIL			
183.	MAILDIR	maildir2mbox maildirwatch	used	These programs use the MAILDIR environment to select the Maildir
184.	MAILTMP	maildir2mbox	used	Temporary file that maildir2mbox can overwrite
185.	MCDFILE	InLookup vuserinfo dbinfo	used	Mail Control Definition file. Specifies the list of all MySQL servers participating in a clustered setup.
186.	MYSQL_TABLE		used	Name of the default indimail authentication table containing active users. If not set, this defaults to '***indimail***'
187.	MYSQL_INACTIVE_TABLE		used	Name of the default indimail authentication table containing inactive users. If not set, this defaults to '***indibak***'

	Name	*Program*	*Set/Used*	*Purpose*
188.	MAXADDR	chowkidar	used	Number of addresses to load in chowkidar
189.	TABLE_NAME	osh	used	Access control file for granting program specific access to users.
190.	TCPLOCALHOST	tcpclient tcpserver	set	The domain name of the local host. If there is no currently available domain name for the local IP address, TCPLOCALHOST is not set
191.	TCPLOCALIP	tcpclient tcpserver	set	The IP address of the local host in dotted-decimal form
192.	TCPLOCALPORT	tcpclient tcpserver	set	The local TCP port numer, in decimal
193.	TCPREMOTEHOST	tcpclient tcpserver tcprulescheck	set	The domain name of the remote host, with uppercase letters converted to lowercase. If there is no currently available domain name for the remote IP address, TCPREMOTEHOST is not set.
194.	TCPREMOTEIP	tcpclient tcpserver tcprulescheck rblsmtpd	set	The IP address of the remote host.
195.	TCPREMOTEPORT	tcpclient tcpserver	set	The remote TCP port number.
196.	TCPREMOTEINFO	tcpclient tcpserver tcprulescheck	set	A connection-specific string, perhaps a username, supplied by the remote host via 931/1413/IDENT/TAP. If the remote host did not supply connection information, TCPREMOTEINFO is not set.
197.	TERM	osh logtop	used	Initialize various escape sequence for terminal handling
198.	GREYIP	qmail-smtpd	used	If set, will send a UDP packet to greydaemon for greylisting
199.	GREETDELAY	qmail-smtpd	used	Time in seconds by which qmail-smtpd delays the “220 ESMTP” welcome prompt. This overrides the greetdelay control file.

	Name	Program	Set/Used	Purpose
200.	SERVERCERT	qmail-smtpd	used	File having SSL certificate to be presented to clients in TLS encrypted sessions. This overrides the control file servercert.pem
201.	CLIENTCA	qmail-smtpd	used	A file having list of Certifying Authority (CA) certificates that are used to verify the client-presented certficates during a TLS-encrypted session. This overrides the control file clientca.pem
202.	CLIENTCRL	qmail-smtpd	used	File having a list of Revocation Lists (CRLs). This overrides the control file clientcrl.pem

Table 1Default Values of Environment Variables

Name of Environment Variable	Default Values or Description
BULK_MAILDIR	bulk_mail
DEFAULT_DOMAIN	set by ./configure
BASE_PATH	/mail
MYSQL_HOST	localhost
MYSQL_VPORT	3306
MYSQL_SOCKET	/tmp/mysql.sock
MYSQL_DATABASE	indimail
MYSQL_TABLE	indimail
MYSQL_INACTIVETABLE	indibak
MYSQL_PASSWD	
MYSQL_USER	indimail
QMAILDIR	/var/indimail
TCP_FILE	/var/indimail/etc/tcp.smtp
OPEN_SMTP_CUR_FILE	/var/indimail/etc/open-smtp
RELAY_TABLE	relay
MCDFILE	"mcdinfo" or /var/indimail/control/mcdinfo
INFIFO	infifo or /var/indimail/control/inquery/infifo

3.3. Relaying

3.3.1. Introduction

What is relaying? It's when an MTA accepts a message via SMTP that doesn't appear to be either for a local address or from a local sender.

In the pre-spam days, it was common for MTA's to be configured as open relays: promiscuous servers that would accept mail from anyone, for anyone.

Most MTA's now are configured to either completely disable relaying, or to only a allow certain trusted users or systems to use them as a relay.

3.3.2. Disabling relaying

If you follow the official directions for installing IndiMail, relaying will be turned off by default. This is accomplished by populating the file /var/indimail/control/rcpthosts with the fully-qualified domain names listed in locals and virtualdomains (the local hosts). The name of the control file, rcpthosts, comes from the SMTP RCPT (recipient) command. In an SMTP session, RCPT is used to specify the addresses of the recipients of a message. rcpthosts, then, lists the valid hostnames that can appear in a RCPT address.

3.3.3. Allowing selective relaying

Most single-user and small workgroup servers can disable relaying completely, but if you have to support a distributed user community, you'll need a way to allow your users, and only your users, to use your system as a relay. This is accomplished by using tcpserver to set the RELAYCLIENT environment variable, which tells qmail-smtpd to override the rcpthosts file.

If you follow the installation instructions in this document, selective relaying will be enabled by default. To give a client relay access, add an entry to /etc/tcp.smtp like:

```
IP address of client:allow,RELAYCLIENT=""
```

Then rebuild the SMTP access database by doing:

```
qmailctl cdb
```

or:

```
tcprules /etc/tcp.smtp.cdb /etc/tcp.smtp.tmp < /etc/tcp.smtp
chmod 644 /etc/tcp.smtp*
tcprules /etc/tcp.imap.cdb /etc/tcp.imap.tmp < /etc/tcp.imap
```

```
chmod 644 /etc/tcp.imap*
tcprules /etc/tcp.pop3.cdb /etc/tcp.pop3.tmp < /etc/tcp.pop3
chmod 644 /etc/tcp.pop3*
```

3.4. Multiple host names

If your system is known by more than one name, e.g., all addresses of the form user@host1.example.com can also be written as user@example.com or user@mail.example.com, then you need to tell qmail this so it'll know which addresses it should deliver locally and which messages it should accept from remote systems.

To do this, just add all of the names to two control files:

- rcpthosts, which tells qmail-smtpd to accept mail addressed to these hosts, and
- locals, which tells qmail-send that addresses on these hosts are to be delivered locally.

Send qmail-send a HUP (hangup) signal to tell it to reread locals. If you have qmailctl, you can do:

```
qmailctl reload
```

3.5. Virtual domains

Virtual domains are similar to the multiple host names discussed in the previous section, but there are some important differences. First, if example.net hosts the virtual domain virtual.example.com, it's generally not true that messages sent to joe@example.net should end up in the same mailbox as messages sent to joe@virtual.example.com. The namespace for each virtual domain is distinct.

With qmail, virtual domains are configured in the virtualdomains file, which consists of one or more entries of the form:

```
user@domain:prepend
```

qmail converts user@domain to prepend-user@domain and treats the result as if domain was local (i.e. *prepend* is a local user). The user@ part is optional. If it's omitted, the entry matches all @domain addresses.

Returning to the example scenario above, if the example.net mail administrator wanted to create a virtual domain, virtual.example.com, under the administrative control of user john, the following entry in virtualdomains would accomplish that:

```
virtual.example.com:john
```

An incoming message to joe@virtual.example.com would be rewritten as john-joe@virtual.example.com and delivered locally. See the .qmail section, and the extension addresses subsection for more information about how john can manage his virtual domain.

As with multiple host names, all virtual domains must be listed in rcpthosts so qmail-smtpd will know to accept messages addressed to them. However, unlike multiple host names, virtual domains must not be added to locals.

After modifying virtualdomains, send qmail-send a HUP (hangup) signal to tell it to reread the file. If you have qmailctl, you can do:

```
/var/indimail/bin/qmailctl reload
```

If you are using supervise, you can do:

```
/var/indimail/bin/svc -h /service*/qmail-send*
```

IndiMail provides a complete suite of programs to manage large number of virtual domains with millions of users in a typical ISP or a Mail Service Provider setup. The section on IndiMail and Virtual Domains details the tools.

3.5.1 Using the alias user

If you don't want to create a new user to handle a virtual domain's mail, you can use the **alias** user, which is created when you install qmail, to handle it instead.

Again, you have a machine called **bedrock.stoneage.com**, and you want to want to handle mail for the virtual domain **flintstone.stoneage.com**. You decide to let the **alias** user handle the mail. In the virtualdomains file on **bedrock.stoneage.com** put:

```
flintstone.stoneage.com:alias-flintstone
```

Now, all mail sent to *user@flintstone.stoneage.com* gets remapped to *alias-flintstone-user@bedrock.stoneage.com*. This means that all mail to the **flintstone.stoneage.com** domain is controlled by the *~alias/.qmail-flintstone-** files on bedrock.stoneage.com. For example, mail to:

```
pebbles@flintstone.stoneage.com
```

would be controlled by *~alias/.qmail-flintstone-pebbles*.

Don't forget to add virtual domains to rcpthosts, too.

To administer virtual domains, IndiMail provides numerous commands and tools. Refer to Chapter 5.8 for more details.

Note: Domain name server (DNS) mail exchanger (MX) records must be set up to direct messages for virtual domains to the appropriate mail server. This is a job for the name server administrator and is beyond the scope of this guide.

3.6. Aliases

qmail's standard aliasing mechanism is a natural outgrowth of qmail's local delivery mechanism. qmail-local attempts to deliver a message addressed to localpart@host to a local user named localpart. If no matching user is found, the message is delivered to the alias user, a pseudo-user on all qmail systems whose home directory is usually /var/indimail/alias.

For example, say you want to create an info@example.com alias that forwards messages to user tom. On example.com, do, as user root:

```
echo \&tom > /var/indimail/alias/.qmail-info
```

The .qmail section and extension addresses subsection describe how to create .qmail files that specify which aliases exist, and what to do with messages sent to them.

The Gotchas appendix covers a couple of tricky cases regarding the usage of alias--aliases containing uppercase characters and dots ('.')--and man dot-qmail contains complete documentation of the usage of .qmail files.

Note that because of the way aliases are implemented in qmail, an alias can never override a valid user's deliveries. E.g., if rachel is a normal user, ~alias/.qmail-rachel will not be used.

The fastforward package provides an alternative aliasing mechanism that puts multiple aliases in a single file compatible with Sendmail's alias database.

The next section, qmail-users, describes another mechanism that can be used to implement aliases.

3.7. qmail-users

qmail-users is a system for assigning addresses to users. A series of configuration files resides under /var/indimail/users. The assign file is a table of assignments. There are two kinds of assignments: simple and wildcard.

Note: assign contains a series of assignments, one per line, followed by a line containing a single dot (.). If you create assign manually, don't forget the dot line.

3.7.1. Simple assignment

A simple assignment looks like:

=address:user:uid:gid:directory:dash:extension:

What this means is that messages received for address will be delivered as user user, with the specified uid and gid, and the file directory/.qmaildashextension will specify how the messages are to be delivered.

3.7.2. Wildcard assignment

A wildcard assignment looks like:

+prefix:user:uid:gid:directory:dash:prepend:

What this means is that messages received for addresses of the form prefixrest will be delivered as user user, with the specified uid and gid, and the file directory/.qmaildashprependrest will specify how the messages are to be delivered.

3.7.3. qmail-user programs

qmail-user has two helper programs: qmail-newu and qmail-pw2u.

qmail-newu processes the assign file and generates a constant database (CDB) file called cdb in /var/indimail/users. CDB is a binary format that can be accessed quickly by qmail-lspawn, even when there are thousands of assignments.

qmail-pw2u converts the system user database, /etc/passwd, into a series of assignments suitable for assign. qmail-pw2u uses a set of files to modify the translation rules.

- include: users to include
- exclude: users to exclude
- mailnames: alternative "mailnames" for users
- subusers: extra addresses handled by a user, with an optional .qmail extension
- append: miscellaneous assignments

Note: If you use qmail-pw2u, don't forget to re-run qmail-pw2u and qmail-newu whenever you add users, remove users, or change UID's or GID's. A typical sequence would be:

```
qmail-pw2u </etc/passwd >/var/indimail/users/assign
qmail-newu
```

3.8. Spam Control

IndiMail has the following control files which can be used for spam control.

1. **badmailfrom**

   This file contains unacceptable sender addresses. A line in ***badmailfrom*** may be of the form @host, meaning every address at host.

   To automate building of ***badmailfrom/badrcptto/spamdb***, the utility **chowkidar** can be used in cron.

```
24 0,4,8,12,16,20 * * * /var/indimail/bin/chowkidar -f /var/log/qmail/deliver.25/current -o
/var/indimail/control/badmailfrom -n 300 -B >/dev/null 2>&1
24 0,4,8,12,16,20 * * * /var/indimail/bin/chowkidar -f /var/log/qmail/deliver.25/current -o
/var/indimail/control/badrcptto -n 300 -T >/dev/null 2>&1
24,54 * * * * /var/indimail/bin/chowkidar -f /var/log/qmail/smtpd.25/current -o
/var/indimail/control/spamdb -n 1 -S >/dev/null 2>&1
```

   Additionall, if you are using a clustered domain, you can run **chowkidar's** synchronize mode of operation. The sync operation will synchronize the ***badmailfrom/badrcptto/spamdb*** control file across all hosts running **chowkidar**.

```
5 0 * * * /var/indimail/bin/chowkidar -r -B -q >/dev/null 2>&1
5 0 * * * /var/indimail/bin/chowkidar -r -T -q >/dev/null 2>&1
25,55 * * * * /var/indimail/bin/chowkidar -r -S -q >/dev/null 2>&1
```

   The first set of two cron entries will add senders to the list of spammers, who send mails in excess of 300 mails/day and spammers who get caught by **bogofilter**. The second set of cron entries will synchronize the local ***badmailfrom/badrcptto/spamdb*** copy with a master copy maintained on ***hostcntrl***. Refer Section for more details on **chowkidar**.

2. **spamignore**

   This contains list of Email-IDs not to be treated as spammers by the AutoSpam control utility chowkidar. A line in ***spamignore*** may be of the form @host, meaning every address at host. It can also be a regular shell wildcard pattern. e.g.

```
update*@bigupdates.com
[0-9][0-9][0-9][0-9].com
answerme@save*
*%*
```

3. **badmailpatterns**

Gives **qmail-smtpd** the ability to filter E-Mails by comparing the sender address with a REGEX pattern. e.g.

*@earthlink.net
!fred@earthlink.net
[0-9][0-9][0-9][0-9].com
answerme@save*
%

4. blackholedsender

qmail-smtpd will exit the SMTP connection without peer (SMTP client) notification after recognizing an envelope sender address listed here. A line may be of the form @host, meaning every address at host.

5. blackholedpatterns

Gives **qmail-smtpd** the ability to filter E-Mails by comparing the sender address with a REGEX pattern. e.g.

*@earthlink.net
!fred@earthlink.net
[0-9][0-9][0-9][0-9].com
answerme@save*
%

6. badrcptto

This file contains unacceptable envelope recipient addresses. A line in ***badrcptto*** may be of the form @host, meaning every address at host.

7. badrcptpatterns

employs the same filtering logic as for ***badmailpatterns***. It allows **qmail-smtpd** to reject SPAM E-Mails including the signature

\[dd.dd.dd.dd\]

in badrcptpatterns file, where dd.dd.dd.dd is the IP address.

8. maxrecipients

maxrecipients is the number of RCPT TO:'s qmail-smtpd will accept in a SMTP session.

Default: 0 which means no restriction. The environment variable MAXRECIPIENTS can be used instead.

9. tarpitcount

tarpitcount ist the number of RPCT TO: qmail-smtpd accepts before it starts tarpitting.
Default: 0 which means no tarpitting. The environment variable TARPITCOUNT can be used instead.

10.tarpitdelay

tarpitdelay is the time in seconds of delay to be introduced after each subsequent RCPTTO:.
Default: 5. The environment variable TARPITDELAY can be used instead.

11.badhelo

Unacceptable HELO/EHELO greeting strings. qmail-smtpd will reject every connection attempt if the client's MTA's HELO/EHLO greeting compares with a REGEX pattern provided in badhelo. To enable badhelo checking, the environment variable BADHELOCHECK should be set.

12.bogofilter

This uses a very powerful technique based on bayesian algorithm. To use it you need to configure few environment variables in the qmail-smtpd run file and setup up an initial spam database. Since bogofilter has self learning capability, you need to set up a feedback loop by creating four email ids. The simple steps are given below

1. Setup up bogofilter in smtpd
   SPAMFILTER=/var/indimail/bin/bogofilter
   SPAMFILTERARGS="-p -d /var/indimail/etc"
   LOGFILTER=1
   MAKE_SEEKABLE=1

If you want to reject spam mails, you can set up the two additional environment variables in the smtpd run filename

   REJECTSPAM=1
   SPAMEXITCODE=0

(Setting REJECTSPAM=2 will blackhole SPAM mails)

2. Setup email ids for feedback loop.

   vadduser register-spam@domain.com pass
   vadduser register-nonspam@domain.com pass
   vadduser spam@domain.com pass

vadduser nonspam@domain.com pass

Activate the above users. i.e.

vmoduser -n register-spam@domain.com
vmoduser -n register-nonspam@domain.com
vmoduser -n spam@domain.com
vmoduser -n nonspam@domain.com

Setup alias to run the program bogofilter-qfe. i.e.
valias -i "| /var/indimail/bin/bogofilter-qfe" register-spam@domain.com
valias -i "| /var/indimail/bin/bogofilter-qfe" register-nonspam@domain.com
valias -i "| /var/indimail/bin/bogofilter-qfe" spam@domain.com
valias -i "| /var/indimail/bin/bogofilter-qfe" nonspam@domain.com

3. Setup the spam database.

   Have a collection of at least 2000 Spam Mails lying in some Maildir

   a) Create Non Spam wordlist
   echo /mail/nonspam/Maildir | bogofilter -d /var/indimail/etc -b -n -v

   b) Create Spam wordlist
   echo /mail/spam/Maildir | bogofilter -d /var/indimail/etc -b -s -v

3.8. Virus Scanning

IndiMail has two built in virus scanners

1. qscanq – qmail virus scanner. It replaces qmail-queue. It initiates a scan on the incoming email, and returns the exit status of the scanner or qmail-multi/qmail-queue to the caller. qscanq can call any scanner defined by the environment variable SCANCMD. qscanq uses ripmime to detach attachments.

2. QHPSI – Qmail High Performance Scanner Interface. This interface allows qmail-queue to call any scanner defined by the QHPSI environment variable. The requirement from the scanner being the ability to detect virii in the base64 content itself (eliminating the need for using utilities like ripmime to detach attachments).

Serious scanning requires a separate virus scanner--either one of the supported commercial scanners or Tomasz Kojm's free Clam Antivirus scanner, available from http://www.clamav.net/. If you have followed the INSTALL instructions, you would have installed support for Clam Antivirus Scanner.

3.9. Accounting

The qmailanalog package provides accounting factility. It consists of a processor multilog-matchup and x* and z* scripts. To setup up accounting, series of steps need to be carried out.

The matchup/multilog-matchup program monitors qmail's delivery attempts. It prints various information in a form suitable for further analysis.

If you have saved the complete qmail log, you can simply run matchup with the log as input, and use the matchup output as described below in Steps 1. and Step 2.

Running matchup again and again on a growing log is unnecessarily slow. You can save time by breaking the log into a series of chunks and running matchup just once on each chunk. (You can also save space by discard-ing a chunk once matchup is done with it.) Note that matchup keeps track of messages and delivery attempts across chunks; see the matchup man page for the correct command lines.

You could pipe qmail's log directly through matchup before it is logged, Beware, however, that matchup will quit if it runs out of memory.

1. Run **qmail-send**'s log through **multilog-matchup**.

For the first time run the following.

```
cat /var/log/qmail/deliver.25/current | /var/indimail/bin/multilog-matchup > out.1 5>pend.1
```

The above command creates output in a form suitable for accounting scripts in the file out.1. Pending deliveries are stored in the file pend.1 which can be used in the next invocation of multilog-matchup. i.e.

```
mv pend.1 pend.cur
cat /var/log/qmail/deliver.25/current pend.cur | /var/indimail/bin/multilog-matchup > out.1 5>pend.1
```

2. The file out.1 should be fed to the following z* scripts to get the accounting information. Each script explains its output. Use zoverall for basis statistics, zsendmail for a sendmail-flavoured log.

   The x* scripts extract information about particular messages, senders, or recipients. You can feed the x* output through the z* scripts. Following is the list of the x* and the z* scripts.

   - xpq
   - xrecipient
   - xsender
   - zddist

- zdeferrals
- zfailures
- zoverall
- zrecipients
- zrhosts
- zrxdelay
- zsenders
- zsendmail
- zsuccesses
- zsuids
- zsmtp
- zspam

You can also run the command /var/indimail/sbin/svctool --report=<zscript> where zscript is one of the above listed z* script

/var/indimail/sbin/svctool --report=zoverall
/var/indimail/sbin/svctool --report=zsenders

/var/indimail/sbin/svctool --report=all runs reports for all the z* script.

4. Usage

This section covers the usage of IndiMail by normal users. If you read or send mail on a IndiMail system, this is where you'll find information about how to do that with IndiMail.

4.1. .qmail files

Delivery of a user's mail is usually controlled by one or more ".qmail" (pronounced dot kyoo mail) files--files in the user's home directory with names beginning with .qmail. The dot-qmail man page describes .qmail file usage.

.qmail files contain a list of delivery instructions, one instruction per line. The first character of the line determines what kind of delivery is involved:

Character	*Deliver Type*	*Value*
#	none (comment)	ignored
|	program	command to be run by shell
/ or .	mbox (if last char isn't a /)	pathname of mbox (including the / or .)
/ or .	maildir (if last char is a /)	pathname of maildir (including the / or .)
&	forward	address to forward message
letter or number	forward	address to forward message (including the first char)

4.1.1. program delivery

When a program delivery instruction is encountered, qmail starts a shell (/bin/sh) to execute the command and feeds the command a copy of the incoming message on standard input. The qmail-command man page documents the details of this process.

Program delivery is very powerful, and can be used to implement a wide range of functionality such as message filtering, automatically responding to messages, and delivery via third-party delivery agents such as procmail.

E.g.:

```
|preline /usr/ucb/vacation mgb
```

This causes qmail to start preline, pass it /usr/ucb/vacation and mgb as arguments, and provide a copy of the message on standard input.

4.1.2. mbox delivery

Mbox is the standard UNIX mailbox format, in which multiple messages are stored in a single file and messages are headed with a "From " line. This line looks like a

header field, but it isn't one: it's just something the delivery agent adds so mail readers can tell where each message begins.
e.g.:

```
./Mailbox
```

This causes messages to be appended to $HOME/Mailbox, with a "From " line prepended. A simple mbox mailbox with a single message looks like:

```
From user1@example.net Thu May 13 18:34:50 1999
Received: (qmail 1287205 invoked from network); 13 May 1999 18:34:49 -0000
From: user1@example.net
To: user2@example.com
Subject: hey

What's up?
```

The first line was added at delivery by qmail.

4.1.3. maildir delivery

Maildir is a mailbox format created to address the shortcomings of the mbox format. A maildir mailbox is a directory containing three subdirectories, new, cur, and tmp. Each message in a maildir mailbox is in a separate file in one of the subdirectories, depending upon its status: new is for unread messages, cur is for messages that have been seen, and tmp is for messages in the process of being delivered. The maildir man page describes the format of a maildir in detail.

One of the benefits of the maildir format is that, even though it doesn't use locking to prevent simultaneous updates from different delivery agents, it's reliable. This means maildir mailboxes can safely reside on NFS-mounted filesystems.

E.g.:

```
./Maildir/
```

This causes messages to be saved in $HOME/Maildir, a maildir-format mailbox.

Note: qmail-local can deliver mail to maildir mailboxes, but it can't create them. Maildir mailboxes should be created with the maildirmake program that comes with qmail. E.g., "maildirmake ~/Maildir". Be sure to run maildirmake as the owner of the maildir, not as root. Your useradd or adduser command might support a "skeleton" directory, e.g. /etc/skel, where you can create a maildir that will be copied for all new users.

4.1.4. forward delivery

Forward deliveries causes the message to be resent to the specified address. Addresses specified in .qmail files can't contain comment fields or extra spaces.

These are wrong:

&<user@example.com>
& user@example.com
&Joe User <user@example.com>

These are correct:

&user@example.com
user@example.com
&user

The first two cause user@example.com to receive a copy of the message. The last sends a copy to the local user user.

4.1.5. extension addresses

qmail supports user-controlled extension addresses. In addition to the base address, username@hostname.domain, users can receive mail at username-extension@hostname.domain. For the remainder of this section, I'll leave off the "@hostname.domain" part since we're considering actions that take place on the local system.

The delivery instructions for username-extension are in ~username/.qmail-extension.

For example, mbhangui-indimail@phoenix.indi.com is controlled by ~mbhangui/.qmail-indimail on host phoenix.

Extensions can have multiple fields, e.g., mbhangui-list-qmail, controlled by ~mbhangui/.qmail-list-qmail. In this example, mbhangui-list-qmail is subscribed to the qmail mailing list, and ~mbhangui/.qmail-list-qmail files the list messages in a separate mailbox.

.qmail files can be wildcarded using -default. So mbhangui-list-qmail could also be handled by ~mbhangui/.qmail-list-default. This would allow one catch-all .qmail file to handle all mbhangui-list-whatever addresses.

Note that mbhangui-list wouldn't be handled by ~mbhangui/.qmail-list-default because it doesn't match the "-" after "list".

qmail uses the closest match it finds. E.g., when a message comes in addressed to mbhangui-list-qmail, it'll use the first one of the following that it finds:

.qmail-list-qmail
.qmail-list-default
.qmail-default

If no matching .qmail file is found, the delivery fails and the message bounces back to the sender.

4.2. Sending messages

Mail users usually don't use the MTA directly to send messages. Typically, messages are composed and sent using a Mail User Agent (MUA) such as pine or mutt, which then calls the MTA to deliver the message. The process of handing a message to the MTA is called injection.

There are two ways to inject messages into most MTA's: via the Simple Mail Transfer Protocol, SMTP, or using a program provided by the MTA for that purpose.

4.2.1. SMTP

MUA's can open a TCP connection to port 25 - the standard SMTP port or port 587 – the standard message submission port on the local host or a designated mail server. The MUA and the MTA then engage in a dialogue that results in either:

- the message being transfered to the MTA, or
- a error status being returned to the MUA

SMTP has no mechanism for authentication, so no username or password is required to send a message. However, many MTA's refuse to accept messages that don't appear to be either from or for a local user. If a properly formatted message is rejected, relaying restrictions are the most likely cause. See the Relaying section for more information about relay configuration.

4.2.2. /var/indimail/bin/sendmail

For many years, Sendmail was the UNIX MTA. It was so ubiquitous, that many programmers just assumed that it was the MTA. As a result, Sendmail's local injection mechanism became the standard Application Programmer's Interface (API) for local mail injection. qmail and other non-Sendmail MTA's provide a sendmail program that works the same way as the real Sendmail's sendmail for local injection.

The qmail sendmail, which is normally in /var/indimail/bin/sendmail, usually replaces the Sendmail sendmail on IndiMail systems. Typical locations of the sendmail program include:

- /usr/lib/sendmail
- /usr/sbin/sendmail

On a typical system, "ls -l path-to-sendmail" should show that sendmail is a symbolic link to /var/indimail/bin/sendmail:

```
 $ ls -l /usr/lib/sendmail
lrwxrwxrwx   1 root     root          29 Feb 19 11:04 /usr/lib/sendmail -> /var/indimail/bin/sendmail
```

The sendmail man page provided with qmail describes how to use the program.

4.2.3. qmail-inject

In addition to emulating the sendmail API, qmail has its own injection program: qmail-inject. In fact, sendmail is just a wrapper around qmail-inject.

As an API, sendmail is probably better because it's much more widely available. The qmail API provided by qmail-inject will only work on systems with qmail, but the sendmail interface is nearly universal.

For example, to send a blank message to joe@example.com:

```
 echo To: joe@example.com | /var/indimail/bin/qmail-inject
```

QMAILINJECT Flags

Letter	Purpose
c	Use address-comment style for the From field
s	Do not look at any incoming Return-Path field
f	Delete any incoming From field
i	Delete any incoming Message-ID field
r	Use a per-recipient VERP
m	Use a per-message VERP

4.3. Architecture

4.3.1 Modular system architecture

Internet MTA's perform a variety of tasks. Earlier designs like Sendmail and smail are monolithic. In other words, they have one large, complex program that "switches hats": it puts on one hat to be an SMTP server, another to be an SMTP client, another

to inject messages locally, another to manage the queue, etc.

qmail is modular. Each of these functions is performed by a separate program. As a result, the programs are much smaller, simpler, and less likely to contain functional or security bugs. To further enhance security, qmail's modules run with different privileges, and they don't "trust" each other: they don't assume the other modules always do only what they're supposed to do.

The core modules are:

Modules	Function
qmail-smtpd	accepts/rejects messages via SMTP
qmail-inject	injects messages locally
qmail-rspawn/qmail-remote	handles remote deliveries
qmail-lspawn/qmail-local	handles local deliveries
qmail-send	processes the queue
qmail-clean	cleans the queue

There's also a down side to the modular approach. Unlike a monolithic MTA, the interactions between modules are well-defined, and modules only exchange the minimum necessary information with each other. This is generally A Good Thing, but sometimes it makes it hard to do things. For example, the sendmail "-v" flag causes Sendmail to print a trace of its actions to standard output for debugging purposes. Since the one sendmail binary handles injection, queueing, alias processing, .forward file processing, and remote forwarding via SMTP, it is able to easily trace the entire delivery until the message is delivered. The equivalent capability in qmail doesn't exist, and would require substantial code changes and additional complexity to implement the passing of the "debug" flag from module to module.

4.3.2. File structure

/var/indimail is the root of the qmail file structure. This can be changed when qmail is being built, but it's a good idea to leave it unchanged so other administrators know where to find things. If you really want to relocate some or all of the qmail tree, it's better to do that using symbolic links. See the Create directories subsection of the Installation section for details.

The top-level subdirectories are:

Directory	Contents
alias	.qmail files for system-wide aliases
bin	program binaries and scripts
boot	startup scripts
control	configuration files
doc	documentation (except man pages)
man	man pages
users	the qmail-users database files

4.3.3 Queue structure

The file INTERNALS in the build directory discusses the details of queueing more thoroughly. This is a broader overview of structure of the queue.

Subdirectory	Contents
bounce	permanent delivery errors
info*	envelope sender addresses
intd	envelopes under construction by qmail-queue
local*	local envelope recipient addresses
lock	lock files
mess*	message files
pid	used by qmail-queue to acquire an i-node number
remote*	remote envelope recipient addresses
todo	complete envelopes

Note: Directories marked with an "*" contain a series of split subdirectories named "0", "1", ..., up to (conf-split-1), where conf-split is a compile-time configuration setting contained in the file conf-split in the build directory. It defaults to 23. The purpose of splitting these directories is to reduce the number of files in a single directory on very busy servers. conf-split must be a prime number.

Files under the mess subdirectory are named after their i-node number. What this means is that you can't manually move them using standard UNIX utilities like mv, dump/restore, and tar. There are a couple user-contributed utilities on http://www.qmail.org that will rename queue files correctly.

Note: It is not safe to modify queue files while qmail is running. If you want to modify the queue, stop qmail first, play with the queue carefully, then restart qmail.

4.3.4 Deferred Message Retry Schedule

Each message has its own retry schedule. The longer a message remains undeliverable, the less frequently qmail tries to send it. The retry schedule is not configurable. The following table shows the retry schedule for a message that's undeliverable to a remote recipient until it bounces. Local messages use a similar, but more frequent, schedule.

Delivery	***Attempts***	***Seconds D-HH:MM:SS***
1	0	0-00:00:00
2	400	0-00:06:40
3	1600	0-00:26:40

Delivery	*Attempts*	*Seconds D-HH:MM:SS*
4	3600	0-01:00:00
5	6400	0-01:46:40
6	10000	0-02:46:40
7	14400	0-04:00:00
8	19600	0-05:26:40
9	25600	0-07:06:40
10	32400	0-09:00:00
11	40000	0-11:06:40
12	48400	0-13:26:40
13	57600	0-16:00:00
14	67600	0-18:46:40
15	78400	0-21:46:40
16	90000	1-01:00:00
17	102400	1-04:26:40
18	115600	1-08:06:40
19	129600	1-12:00:00
20	144400	1-16:06:40
21	160000	1-20:26:40
22	176400	2-01:00:00
23	193600	2-05:46:40
24	211600	2-10:46:40
25	230400	2-16:00:00
26	250000	2-21:26:40
27	270400	3-03:06:40
28	291600	3-09:00:00
29	313600	3-15:06:40
30	336400	3-21:26:40
31	360000	4-04:00:00
32	384400	4-10:46:40
33	409600	4-17:46:40
34	435600	5-01:00:00
35	462400	5-08:26:40
36	490000	5-16:06:40
37	518400	6-00:00:00
38	547600	6-08:06:40
39	577600	6-16:26:40

Delivery	*Attempts*	*Seconds D-HH:MM:SS*
40	608400	7-01:00:00

4.3.5. Large Sites with lots of MX

If you're getting:

deferral: CNAME_lookup_failed_temporarily._(#4.4.3)/

The problem might be that qmail can't handle large name server query responses. The fix is to install djbdns. See Patches under Advanced Topics.

There's also a question as to why some people don't have trouble reaching such systems. Basically, depending on the timing and ordering of queries made to your local nameserver, the size of the response to an ANY query for "aol.com" may be larger than the 512 byte limit of a UDP packet, or it may not.

"May not" is likely to happen if the A and MX records time out, but the NS records don't. Since the .COM servers set a 2 day TTL on those, but AOL sets a 1 hour TTL on their records, this will often happen on less busy nameservers. Busier nameservers are more likely to have those records in their cache at any given time, frustrating an unpatched qmail's attempts to check for CNAMEs.

A better test is to send mail to nosuchuser@large-mx.ckdhr.com; if it clears your queue and winds up bouncing from ckdhr.com, your MTA can send mail to hosts with MX lists that exceed 512 bytes. (By using a single RRset, with a single TTL, that exceeds 512 bytes, the problem can be seen without depending on the timing and ordering of other queries.)

4.3.6. Internals

4.3.6.1 Overview

Here's the data flow in the qmail suite:

qmail-smtpd	-->	qmail-queue	-->	qmail-send qmail-clean	-->	qmail-rspawn	-->	qmail-remote
qmail-inject						qmail-lspawn		qmail-local

Every message is added to a central queue directory by qmail-queue. qmail-queue is invoked as needed, usually by qmail-inject for locally generated messages, qmail-smtpd for messages received through SMTP, qmail-local for forwarded messages, or qmail-send for bounce messages.

Every message is then delivered by qmail-send, in cooperation with qmail-lspawn and

qmail-rspawn, and cleaned up by qmail-clean. These four programs are long-running daemons.

The queue is designed to be crashproof, provided that the underlying filesystem is crashproof. All cleanups are handled by qmail-send and qmail-clean without human intervention. See section 4.3.6.6 for more details.

4.3.6.2. Queue structure

Each message in the queue is identified by a unique number, let's say 457. The queue is organized into several directories, each of which may contain files related to message 457:

- mess/457: the message
- todo/457: the envelope: where the message came from, where it's going
- intd/457: the envelope, under construction by qmail-queue
- info/457: the envelope sender address, after preprocessing
- local/457: local envelope recipient addresses, after preprocessing
- remote/457: remote envelope recipient addresses, after preprocessing
- bounce/457: permanent delivery errors

Here are all possible states for a message.

+means a file exists; -means it does not exist; ? means it may or may not exist.

- S1. -mess -intd -todo -info -local -remote -bounce
- S2. +mess -intd -todo -info -local -remote -bounce
- S3. +mess +intd -todo -info -local -remote -bounce
- S4. +mess ?intd +todo ?info ?local ?remote -bounce (queued)
- S5. +mess -intd -todo +info ?local ?remote ?bounce (preprocessed)

Guarantee: If mess/457 exists, it has inode number 457.

4.3.6.3. Queuing

To add a message to the queue, qmail-queue first creates a file in a separate directory, pid/, with a unique name. The filesystem assigns that file a unique inode number. qmail-queue looks at that number, say 457. By the guarantee above, message 457 must be in state S1.

qmail-queue renames pid/whatever as mess/457, moving to S2. It writes the message to mess/457. It then creates intd/457, moving to S3, and writes the envelope information to intd/457.

Finally qmail-queue creates a new link, todo/457, for intd/457, moving to S4. At that instant the message has been successfully queued, and qmail-queue leaves it for further handling by qmail-send.

qmail-queue starts a 24-hour timer before touching any files, and commits suicide if the timer expires.

4.3.6.4. Queue preprocessing

Once a message has been queued, qmail-send must decide which recipients are local and which recipients are remote. It may also rewrite some recipient addresses.

When qmail-todo notices todo/457, it knows that message 457 is in S4. It removes info/457, local/457, and remote/457 if they exist. Then it reads through todo/457. It creates info/457, possibly local/457, and possibly remote/457. When it is done, it removes intd/457. The message is still in S4 at this point. Finally qmail-todo removes todo/457, moving to S5. At that instant the message has been successfully preprocessed.

4.3.6.5. Delivery of preprocessed messages

Messages at S5 are handled as follows. Each address in local/457 and remote/457 is marked either NOT DONE or DONE.

- DONE: The message was successfully delivered, or the last delivery attempt met with permanent failure. Either way, qmail-send should not attempt further delivery to this address.
- NOT DONE: If there have been any delivery attempts, they have all met with temporary failure. Either way, qmail-send should try delivery in the future.

qmail-send may at its leisure try to deliver a message to a NOT DONE address. If the message is successfully delivered, qmail-send marks the address as DONE. If the delivery attempt meets with permanent failure, qmail-send first appends a note to bounce/457, creating bounce/457 if necessary; then it marks the address as DONE. Note that bounce/457 is not crashproof.

qmail-send may handle bounce/457 at any time, as follows: it

1. injects a new bounce message, created from bounce/457 and mess/457;
2. deletes bounce/457.

When all addresses in local/457 are DONE, qmail-send deletes local/457. Same for remote/457.

When local/457 and remote/457 are gone, qmail-send eliminates the message, as follows. First, if bounce/457 exists, qmail-send handles it as described above. Once bounce/457 is definitely gone, qmail-send deletes info/457, moving to S2, and finally mess/457, moving to S1.

4.3.6.6. Cleanups

If the computer crashes while qmail-queue is trying to queue a message, or while qmail-send is eliminating a message, the message may be left in state S2 or S3.

When qmail-send sees a message in state S2 or S3---other than one it is currently eliminating!---where mess/457 is more than 36 hours old, it deletes intd/457 if that exists, then deletes mess/457. Note that any qmail-queue handling the message must be dead.

Similarly, when qmail-send sees a file in the pid/ directory that is more than 36 hours old, it deletes it.

Cleanups are not necessary if the computer crashes while qmail-send is delivering a message. At worst a message may be delivered twice. (There is no way for a distributed mail system to eliminate the possibility of duplication. What if an SMTP connection is broken just before the server acknowledges successful receipt of the message? The client must assume the worst and send the message again. Similarly, if the computer crashes just before qmail-send marks a message as DONE, the new qmail-send must assume the worst and send the message again. The usual solutions in the database literature---e.g., keeping log files---amount to saying that it's the recipient's computer's job to discard duplicate messages.)

4.3.6.7. Notes

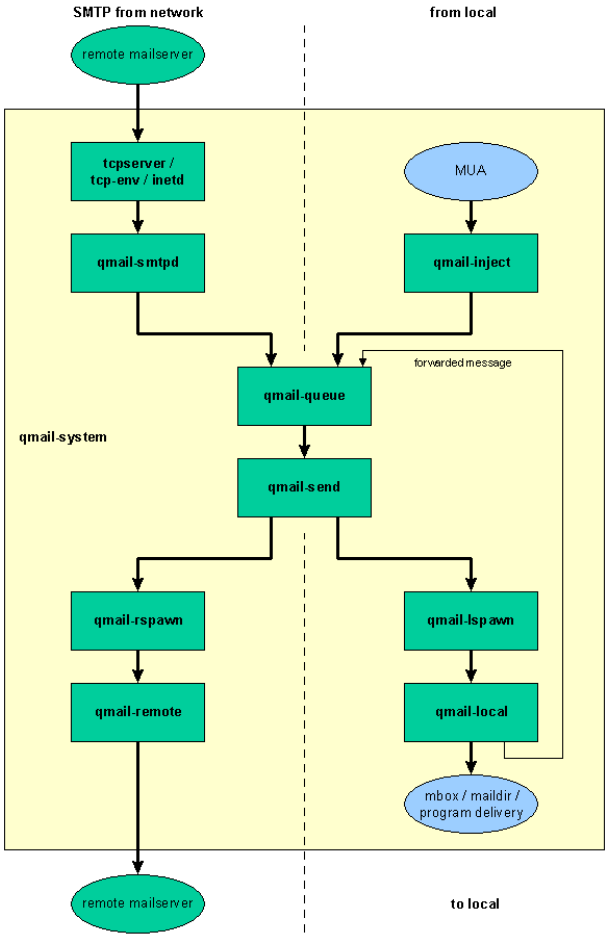
Currently info/457 serves two purposes: first, it records the envelope sender; second, its modification time is used to decide when a message has been in the queue too long. In the future info/457 may store more information. Any non-backwards-compatible changes will be identified by version numbers.

When qmail-queue has successfully placed a message into the queue, it pulls a trigger offered by qmail-send. Here is the current triggering mechanism: lock/trigger is a named pipe. Before scanning todo/, qmail-send opens lock/trigger O_NDELAY for reading. It then selects for readability on lock/trigger. qmail-queue pulls the trigger by writing a byte O_NDELAY to lock/trigger. This makes lock/trigger readable and wakes up qmail-send. Before scanning todo/ again, qmail-send closes and reopens lock/trigger.

THE BIG Qmail PICTURE

Version 1.03 for qmail 1.03

This picture is intended to give a broad overviewof on howall the small qmail programs fit together and to have an quick reference handy. The man pages of the programs give more information on howthings are working in detail.



tcpserver

Command line parameters:
[-qQvdDoOpPhHrR1]
[-climit] [-bbacklog]
[-xrules.cdb] [-ggid] [-uuid]
[-llocalname] [-ttimeout]
host port program [arg ...]

Environment Variables:
none

Configuration files:
rules.cdb

qmail-smtpd

Command line parameters:
none

Environment Variables:
PROTO (TCP)
TCPLOCALHOST
TCPLOCALIP
TCPLOCALPORT
TCPREMOTEHOST
TCPREMOTEINFO
TCPREMOTEIP
TCPREMOTEPORT
RELAYCLIENT

Configuration files:
badmailfrom
databytes
localiphost (me)
rcpthosts
morercpthosts (cdb format)
smtpgreeting (me)
timeoutsmtpd (1200)

qmail-queue

Command line parameters:
none

Environment Variables:
none

Configuration files:
none

Exit codes:
0 success
11 address too long
51 out of memory
52 timeout
53 write error
54 read error
61 problem with qmail homedir
62 problem with queuedir
63 problem with queue/pid
64 problem with queue/mess
65 problem with queue/intd
66 problem with todo
81 internal bug, eg. segfault
91 envelope format error

qmail-rspawn

Command line parameters:
none

Environment Variables:
none

Configuration files:
none

Exit codes:
0 success
100 permanent error (bounce)
111 soft error (retry later)

qmail-remote

Command line parameters:
host
sender
recip [recip ...]

Environment Variables:
none

Configuration files:
helohost (me)
smtproutes
timeoutconnect (60)
timeoutremote (1200)

Exit codes:
always 0 and all errors come back as clear text

.qmail

Content:
#comment
|program to execute
&forwarding email address
./mailbox deliver to Mbox
./maidir/ deliver to Maildir

Exit codes of program:
0 success
99 success and abort .qmail
100 permanent error (bounce)
111 soft error (retry later)

qmail-inject

Command line parameters:
[-nNaAhH]
[-fsender]
recip

Environment Variables:
QMAILUSER
MAILUSER
USER
LOGNAME
QMAILHOST
MAILHOST
QMAILNAME
MAILNAME
NAME
QMAILINJECT
QMAILDEFAULTDOMAIN
QMAILDEFAULTHOST
QMAILIDHOST
QMAILPLUSDOMAIN

Configuration files:
defaultdomain (me)
defaulthost (me)
idhost (me)
plusdomain (me)

qmail-send

Command line parameters:
none

Environment Variables:
none

Configuration files:
bouncefrom (mailer-daemon)
bouncehost (me)
concurrencylocal (10)
concurrencyremote (20)
doublebouncehost (me)
doublebounceto (postmaster)
envnoathost (me)
locals (me)
percenthack
queuelifetime (604800)
virtualdomains

Remarks:
virtualdomains are handled after locals

qmail-lspawn

Command line parameters:
aliasempty

Environment Variables:
none

Configuration files:
~users/cdb
~bin/qmail-getpw

Exit codes:
0 success
100 aliasempty missing
111 soft error (retry later)
Most errors come back as clear text

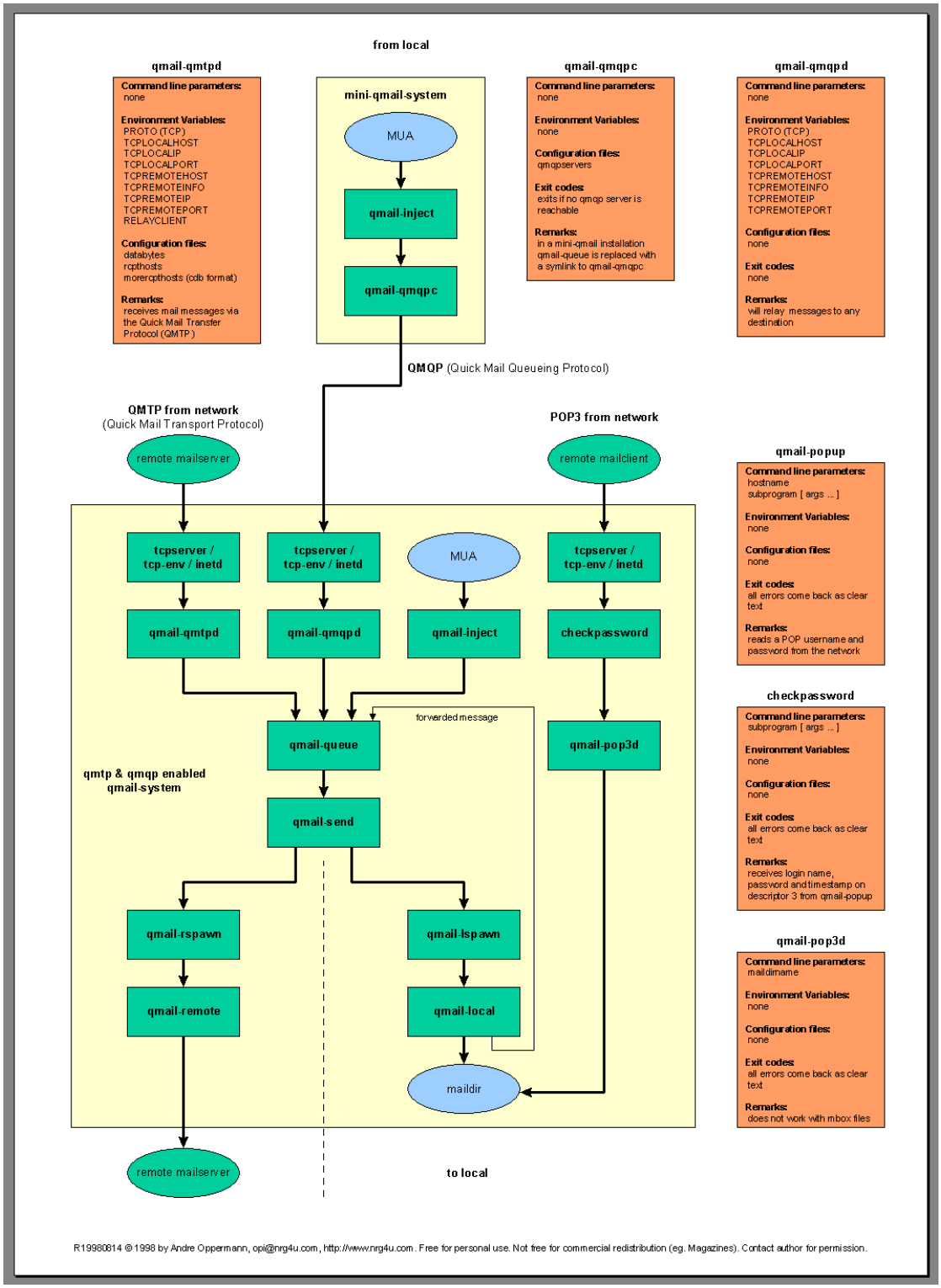
qmail-local

Command line parameters:
[-nN]
user
homedir
local
dash
ext
domain
sender
aliasempty

Environment Variables:
none

Configuration files:
~home/.qmail
~alias/.qmail

Exit codes:
0 success
1 fatal error
2 chdir error
3 SIGALRM
4 tryunlinktmp() error
100 permanent error (bounce)
111 soft error (retry later)
Most errors come back as clear text


from local
qmail-qmtpd
Command line parameters:
none
Environment Variables:
PROTO (TCP)
TCPLOCALHOST
TCPLOCALIP
TCPLOCALPORT
TCPREMOTEHOST
TCPREMOTEINFO
TCPREMOTEIP
TCPREMOTEPORT
RELAYCLIENT
Configuration files:
databytes
rcpthosts
morercpthosts (cdb format)
Remarks:
receives mail messages via the Quick Mail Transfer Protocol (QMTP)
mini-qmail-system
MUA
qmail-inject
qmail-qmqpc
qmail-qmqpc
Command line parameters:
none
Environment Variables:
none
Configuration files:
qmqpservers
Exit codes:
exits if no qmqp server is reachable
Remarks:
in a mini-qmail installation qmail-queue is replaced with a symlink to qmail-qmqpc
qmail-qmqpd
Command line parameters:
none
Environment Variables:
PROTO (TCP)
TCPLOCALHOST
TCPLOCALIP
TCPLOCALPORT
TCPREMOTEHOST
TCPREMOTEINFO
TCPREMOTEIP
TCPREMOTEPORT
Configuration files:
none
Exit codes:
none
Remarks:
will relay messages to any destination
QMQP (Quick Mail Queueing Protocol)
QMTP from network
(Quick Mail Transport Protocol)
remote mailserver
POP3 from network
remote mailclient
tcpserver / tcp-env / inetd
tcpserver / tcp-env / inetd
MUA
tcpserver / tcp-env / inetd
qmail-qmtpd
qmail-qmqpd
qmail-inject
checkpassword
forwarded message
qmail-queue
qmail-pop3d
qmtp & qmqp enabled qmail-system
qmail-send
qmail-rspawn
qmail-lspawn
qmail-remote
qmail-local
maildir
remote mailserver
to local
qmail-popup
Command line parameters:
hostname
subprogram [args ...]
Environment Variables:
none
Configuration files:
none
Exit codes:
all errors come back as clear text
Remarks:
reads a POP username and password from the network
checkpassword
Command line parameters:
subprogram [args ...]
Environment Variables:
none
Configuration files:
none
Exit codes:
all errors come back as clear text
Remarks:
receives login name, password and timestamp on descriptor 3 from qmail-popup
qmail-pop3d
Command line parameters:
maildirname
Environment Variables:
none
Configuration files:
none
Exit codes:
all errors come back as clear text
Remarks:
does not work with mbox files
R19980814 © 1998 by Andre Oppermann, opi@nrg4u.com, http://www.nrg4u.com. Free for personal use. Not free for commercial redistribution (eg. Magazines). Contact author for permission.

dot-qmail programs

bouncesaying

Description:
bounce each incoming message (according to the exit value of [program])

Command line parameters:
in .qmail:
|bouncesaying
error
[program [arg ...]]

Environment Variables:
none

Configuration files:
none

Exit codes of program:
0 bounce and say error
111 soft error (retry later)
all other errors are ignored and the rest of .qmail will be processed as usual

condredirect

Description:
redirect message to another address according to the exit value of program

Command line parameters:
in .qmail:
|condredirect
newaddress
program [arg ...]

Environment Variables:
none

Configuration files:
none

Exit codes of program:
0 forward to newaddress
111 soft error (retry later)
all other errors are ignored and the rest of .qmail will be processed as usual

except

Description:
reverse the exit code of a program

Command line parameters:
in .qmail:
|except
program [arg ...]

Environment Variables:
none

Configuration files:
none

Exit codes of program:
0 except exits 100
111 soft error (retry later)
all other errors are ignored and the rest of .qmail will be processed as usual

forward

Description:
forward message to one or more addresses

Command line parameters:
in .qmail:
|forward
address ...

Environment Variables:
none

Configuration files:
none

Exit codes of program:
none

preline

Description:
prepend UUCP-style lines

Command line parameters:
in .qmail:
|preline
command
[-d] (no Delivered-To line)
[-f] (no From_line)
[-r] (no Return-Path line)

Environment Variables:
none

Configuration files:
none

Exit codes of program:
none

Remarks:
is useful for procmail and ELM's filter

qbiff

Description:
announce newmessage the moment it arrives

Command line parameters:
in .qmail:
|qbiff

Environment Variables:
none

Configuration files:
none

Exit codes:
none

Remarks:
writes a message to your screen whenever a new message is delivered

qlist

Description:
handle mailing list subscription requests

Command line parameters:
in .qmail-list-request
|qlist
read man page for details

Environment Variables:
none

Configuration files:
none

Exit codes:
none

Remarks:
read man page for details

qreceipt

Description:
respond to delivery notice requests

Command line parameters:
in .qmail:
|qreceipt
youraddress

Environment Variables:
none

Configuration files:
none

Exit codes:
none

Remarks:
sends a success notice back to the envelope sender

userland programs

maildirmake

Description:
creates a maildir structure

Command line parameters:
directory

Environment Variables:
none

Configuration files:
none

Exit codes:
none

maildirwatch

Description:
watch for new messages in a maildir

Command line parameters:
you have so set the environment

Environment Variables:
MAILDIR

Configuration files:
none

Exit codes:
none

Remarks:
prints a new mail summary twice per minute

maildir2mbox

Description:
moves messages from maildir to mbox format

Command line parameters:
you have so set the environment

Environment Variables:
MAILDIR
MAIL
MAILTMP

Configuration files:
none

Exit codes:
none

Remarks:
you should run only one maildir2mbox at a time

mailsubj

Description:
send a mail message with a subject line

Command line parameters:
subject
recip ...

Environment Variables:
none

Configuration files:
none

Exit codes:
none

Remarks:
reads the body ofthe message from its standard input

queue management

qmail-qstat

Description:
summarize status of mail queue

Command line parameters:
none

Environment Variables:
none

Configuration files:
none

Exit codes:
complains if there is a problem

Remarks:
must be run either as root or with gid qmail

qmail-qread

Description:
list outgoing messages and recipients

Command line parameters:
none

Environment Variables:
none

Configuration files:
none

Exit codes:
complains if there is a problem

Remarks:
must be run either as root or with uid qmails and gid qmail

qmail-tcpto

Description:
prints qmail-remote's current list of timeouts

Command line parameters:
none

Environment Variables:
none

Configuration files:
none

Exit codes:
complains if there is a problem

Remarks:
must be run either as root or with uid qmailr and gid qmail

qmail-tcpok

Description:
erases qmail-remote's current list of timeouts

Command line parameters:
none

Environment Variables:
none

Configuration files:
none

Exit codes:
complains if there is a problem

Remarks:
must be run either as root or with uid qmailr and gid qmail

Hint: to reschedule every message in the queue for immediate delivery, do a "kill -ALRM *pid*" on the qmail-send pid

user and system management

qmail-pw2u

Description:
build address assignments from a passwd file

Command line parameters:
[-/bhHuUC] [-cchar]

Environment Variables:
none

Configuration files:
~users/include
~users/exclude
~users/mailnames
~users/subusers
~users/append

Exit codes:
complains if there is a problem

Remarks:
generates ~users/assign

qmail-newu

Description:
prepare address assignments for qmail-lspawn

Command line parameters:
none

Environment Variables:
none

Configuration files:
~users/assign

Exit codes:
complains if there is a problem with ~users/assign qmail-newu

Remarks:
generates ~users/cdb

qmail-newmrh

Description:
prepare morercpthosts for qmail-smtpd

Command line parameters:
none

Environment Variables:
none

Configuration files:
~control/morercpthosts

Exit codes:
complains if there is a problem with control/morercpthosts qmail-newmrh complains

Remarks:
generates ~control/morercpthosts.cdb

qmail-showctl

Description:
analyze the qmail config files

Command line parameters:
none

Environment Variables:
none

Configuration files:
all

Exit codes:
complains if there is a problem

Remarks:
explains the current qmail configuration

other qmail daemons

qmail-start

Description:
turn on mail delivery

Command line parameters:
defaultdelivery
logger [args ...]

Environment Variables:
none

Configuration files:
none

Exit codes:
does not print anything, even on failure

Remarks:
make sure to clean up the environment before starting qmail

qmail-clean

Description:
clean up the queue directory

Command line parameters:
none

Environment Variables:
none

Configuration files:
none

Exit codes:
none

Remarks:
can only be started by qmail-start

splogger

Description:
reads a series of messages and feeds them to syslog

Command line parameters:
[tag [facility]]

Environment Variables:
none

Configuration files:
none

Exit codes:
complains if there is a problem

Remarks:
converts unprintable characters to question marks

4.4. Troubleshooting

4.4.1. Processes

A properly-running, complete, but minimal IndiMail installation should always have the following seven processes:

- qmail-send running as user qmails
- qmail-todo running as user qmails
- qmail-clean running as user qmailq
- qmail-rspawn running as user qmailr
- qmail-lspawn running as user root
- qmail-daemon running as user root
- qmail-cat running as user indimail
- inlookup running as user indimail

Depending upon your flavor of UNIX, one of the following two commands should list these processes, and possibly a few more:

```
ps -ef | grep qmail
ps waux | grep qmail
```

For example:

```
$ ps waux|grep qmail
root        1123  0.0  0.3  2140  876 ?        S    09:44   0:00 /bin/sh /var/indimail/bin/svscanboot
root        1195  0.0  0.1  1380  280 ?        S    09:44   0:00 /var/indimail/bin/svscan /service
root        1196  0.0  0.0  1332  216 ?        S    09:44   0:00 /var/indimail/bin/readproctitle /service
errors: ....................................................................................................
....................................................................................................
.
root        1197  0.0  0.1  1340  260 ?        S    09:44   0:00 supervise qmail-send.25
root        1199  0.0  0.1  1340  260 ?        S    09:44   0:00 supervise qmail-smtpd.25
root        1201  0.0  0.1  1340  260 ?        S    09:44   0:00 supervise qmail-imapd.143
root        1203  0.0  0.1  1340  260 ?        S    09:44   0:00 supervise qmail-pop3d.110
root        1217  0.0  0.1  1340  260 ?        S    09:44   0:00 supervise qmail-imapd.4143
root        1219  0.0  0.1  1340  260 ?        S    09:44   0:00 supervise qmail-pop3d.4110
qmaill      1223  0.0  0.1  1352  240 ?        S    09:44   0:00 /var/indimail/bin/multilog t /var/log/qmail/imapd.4143
qmaill      1225  0.0  0.1  1352  240 ?        S    09:44   0:00 /var/indimail/bin/multilog t /var/log/qmail/pop3d.4110
qmaill      1226  0.0  0.1  1352  256 ?        S    09:44   0:00 /var/indimail/bin/multilog t /var/log/qmail/deliver.25
qmaill      1228  0.0  0.1  1352  256 ?        S    09:44   0:00 /var/indimail/bin/multilog t /var/log/qmail/smtpd.25
qmaill      1230  0.0  0.1  1352  256 ?        S    09:44   0:00 /var/indimail/bin/multilog t /var/log/qmail/imapd.143
qmaill      1232  0.0  0.1  1352  240 ?        S    09:44   0:00 /var/indimail/bin/multilog t /var/log/qmail/pop3d.110
qmaill      1245  0.0  0.1  1352  256 ?        S    09:44   0:00 /var/indimail/bin/multilog t /var/log/qmail/InLookup.infifo
qmaill      1249  0.0  0.1  1352  256 ?        S    09:44   0:00 /var/indimail/bin/multilog t /var/log/qmail/indisrvr.4000
indimail 14946  0.0  0.1  1368  308 ?        S    16:28   0:00 /var/indimail/bin/tcpserver -c 40 -C 10 -x
/var/indimail/etc/tcp.imap.cdb -X -o -b 40 -H -l 0 -R -u 508 -g 508 0 143 /var/indimail/sbin/imaplogin
/var/indimail/libexec/authlib/authpam /var/indimail/libexec/authlib/authindi /var/indimail/bin/imapd Maildir

indimail 14957  0.0  0.1  1368  304 ?        S    16:28   0:00 /var/indimail/bin/tcpserver -c 40 -C 10 -o -b 40 -H -l 0 -R -x
/var/indimail/etc/tcp.imap.cdb -X -u 508 -g 508 0 4143 /var/indimail/bin/proxyimap /var/indimail/bin/imapd Maildir

indimail 14968  0.0  0.1  1368  304 ?        S    16:28   0:00 /var/indimail/bin/tcpserver -c 40 -C 10 -x
/var/indimail/etc/tcp.imap.cdb -X -o -b 40 -H -l 0 -R -u 508 -g 508 0 110 /var/indimail/sbin/pop3login
/var/indimail/libexec/authlib/authpam /var/indimail/libexec/authlib/authindi /var/indimail/bin/pop3d Maildir
```

```
indimail 14979  0.0  0.1  1368  304 ?        S    16:28   0:00 /var/indimail/bin/tcpserver -c 40 -C 10 -x
/var/indimail/etc/tcp.imap.cdb -X -o -b 40 -H -l 0 -R -u 508 -g 508 0 4110 /var/indimail/bin/proxypop3 /var/indimail/bin/pop3d
Maildir

root       14990  0.0  0.1  1360  304 ?        S    16:28   0:00 qmail-daemon ./Maildir/
qmails     14992  0.0  0.1  1384  328 ?        S    16:28   0:00 qmail-send
root       14994  0.0  0.3  3680  920 ?        S    16:28   0:00 qmail-lspawn ./Maildir/
qmailr     14995  0.0  0.3  3688  920 ?        S    16:28   0:00 qmail-rspawn
qmailq     14996  0.0  0.1  1348  296 ?        S    16:28   0:00 qmail-clean
qmails     14999  0.0  0.1  1368  312 ?        S    16:28   0:00 qmail-todo
qmailq     15001  0.0  0.1  1348  296 ?        S    16:28   0:00 qmail-clean

indimail 15004  0.0  0.1  1368  308 ?        S    16:28   0:00 /var/indimail/bin/tcpserver -H -R -l 0 -x
/var/indimail/etc/tcp.smtp.cdb -c 50 -o -b 50 -u 508 -g 508 0 25 /var/indimail/bin/qmail-smtpd technology.indimail.org
/var/indimail/bin/vchkpass /var/indimail/bin/systpass /bin/false

indimail 15012  0.0  0.3  3656  940 ?        S    16:28   0:00 /var/indimail/bin/InLookup -f infifo -i 5
indimail 15013  0.0  0.5  3712 1216 ?        S    16:28   0:00 /var/indimail/bin/InLookup -f infifo -i 5
indimail 15014  0.0  0.5  3752 1428 ?        S    16:28   0:00 /var/indimail/bin/InLookup -f infifo -i 5
indimail 15015  0.0  0.5  3712 1216 ?        S    16:28   0:00 /var/indimail/bin/InLookup -f infifo -i 5
indimail 15016  0.0  0.5  3712 1216 ?        S    16:28   0:00 /var/indimail/bin/InLookup -f infifo -i 5
indimail 15017  0.0  0.5  3752 1428 ?        S    16:28   0:00 /var/indimail/bin/InLookup -f infifo -i 5
root       15020  0.0  0.4  3656 1020 ?        S    16:28   0:00 /var/indimail/bin/indisrvr -i 0 -p 4000 -b 50
$
```

If you run qmail or qmail-smtpd under supervise, as in the example above, you should see those processes as well. And if run qmail-smtpd under tcpserver, you should see a parent tcpserver process plus an additional tcpserver process for each active incoming SMTP connection.

If you use splogger (or multilog or cyclog) to handle logging, you'll have a splogger (or multilog or cyclog) process or two running as user qmaill.

Also, if qmail is busy delivering messages locally or remotely, you'll see up to concurrencylocal qmail-local processes and up to concurrencyremote qmail-remote processes.

4.4.2. Logging

IndiMail uses **multilog to** log messages to a series of files in a specified directory.

The log directory is specified on the multilog command line, so you can find it by examining your qmail startup script.

The number of files in the log directory, and the maximum size of each file, are determined by multilog options. The log file names are the TAI (Temps Atomique International) timestamps of the time at which the file was started. The tai64nlocal command, also from daemontools, converts TAI timestamps into local, human-readable timestamps.

A typical multilog log entry looks like:

```
@4000000038c3eeb104a6ecf4 delivery 153: success: did_1+0+0/
```

"@4000000038c3eeb104a6ecf4" is the optional, but recommended, TAI timestamp. "delivery 153: success: did_1+0+0/" is the log message itself.

4.4.2.1. Log messages

Here's a typical log sequence for a message sent to a remote system from the local system:

1. @4000000038c3eeb027f41c7c new msg 93869
2. @4000000038c3eeb027f6b0a4 info msg 93869: bytes 2343 from <mbhangui@indi.com> qp 18695 uid 49491
3. @4000000038c3eeb02877ee94 starting delivery 2392: msg 93869 to remote lwq@w3.to
4. @4000000038c3eeb0287b55ac status: local 0/10 remote 1/20
5. @4000000038c3eeb104a13804 delivery 2392: success: 209.85.127.177_accepted_message. /Remote_host_said:_250_CAA01516_Message_accepted_for_delivery/
6. @4000000038c3eeb104a4492c status: local 0/10 remote 0/20
7. @4000000038c3eeb104a6ecf4 end msg 93869

Line 1 indicates that qmail has received a new message, and its queue ID is 93869. The queue ID is the i-node number of the /var/indimail/queue/mess/NN/ file--the queue file that contains the message. The queue ID is guaranteed to be unique as long as the
message remains in the queue.

Line 2 says that the message is from mbhangui@indi.com and is 189 bytes.

Line 3 says qmail-remote is starting to deliver the message to lwq@w3.to, and it's assigning the ID 2392 to the delivery.

Line 4 says 0 local deliveries and 1 remote delivery are pending.

Line 5 says delivery 2392 is complete and successful, and it returns the remote server's response, which often contains information the remote mail administrator would find helpful in tracking a delivery. In this case, the "CAA01516" is the remote system's delivery ID.

Line 6 says 0 local deliveries and 0 remote deliveries are pending, i.e., the delivery is complete.

Line 7 says that the message has been delivered completely and removed from the queue. At this point, the queue ID, 93869, is reusable for another delivery.

4.5. IndiMail and Virtual Domains

Refer to Chapter 3 on Installation. After the installation, you are now ready to add virtual domains and virtual users and be able to send/receive mails for these users. Following Programs are available to assist you in doing this. You may want to refer to Chapter 3.5 on Virtual Domains

1. vadddomain
2. vmoddomain
3. vaddaliasdomain
4. vatrn
5. vdeldomain
6. vdominfo
7. printdir
8. vmoddomlimits
9. vreorg
10. vsmtp
11. vipmap
12. vcalias
13. vrenamedomain
14. hostsync
15. vadduser
16. vmoduser
17. vgroup
18. vpasswd
19. vdeluser
20. vdeloldusers
21. vsetuserquota
22. vuserinfo
23. valias
24. vmoveuser
25. vrenameuser
26. vchkpass
27. vdelivermail
28. postdel
29. clearopensmtp
30. vbulletin
31. vacation
32. updaterules
33. indiversion
34. versioninfo
35. indisrvr
36. adminclient
37. mgmtpass
38. vpriv
39. vcaliasrev.sh

40.proxyimap
41.proxypop3
42.vproxy
43.vhostid
44.dbinfo
45.ipchange
46.InLookup
47.inquerytest
48.vfstab
49.vcfilter
50.vfilter
51.execmysql
52.chowkidar
53.logmonitor
54.controlsync
55.updatefile
56.hostcntrl
57.resetquota

The chapters below describe the usage of these commands.

vadddomain[+]

We recommend that all domains be setup as virtualdomains. This is done through the progam **vadddomain.**

SYNOPSYS

vadddomain [options] domain_name [postmaster password]

DESCRIPTION

Add a new virtual domain. All qmail files and system are automatically updated. It also adds the RFC821 required ***postmaster*** account. Carries out the following steps to create the domain.

- Create the domains directory (/var/indimail/domains/domain_name)
- Create the *.qmail-default* file in the domains directory.
- Add the domain to qmail *assign* file.
- Add the domain to *rcpthosts*, *virtualdomains* (and *etrnhosts* for domains with ETRN/AUTOTURN support).
- Create table domain_name if indimail has been configured as a *large site*. For a *small site*, creates table ***indimail*** and ***indibak*** if used for the first time.
- Sends SIGHUP to **qmail-send.**
- Add entry to table ***dbinfo*** and ***mcdfile*** for a clustered domain ('**-D**' option).
- Create the **postmaster** account for a non-clustered domain. For a clustered domain, the **postmaster** account is created on the first host on which the domain is created.

+

1. The virtual domain becomes active after running this program. No additional files need to be modified. A new directory is added under ~indimail/domains to house all the .qmail files, configuration files for the new virtual domain.
2. vadddomain has the setuid bit set and runs as root.
3. Domain names can contain alphanumeric characters besides '-' and '.' . However the first and the last character cannot be '-'.
4. The postmaster account is created as an active account.
5. For a clustered domain, the postmaster id is added only for the first domain part of the cluster.

6. The virtual domain becomes active after running this program. No additional files need to be modified. A new directory is added under ~indimail/domains to house all the .qmail files, configuration files for the new virtual domain.
7. vadddomain has the setuid bit set and runs as root.
8. Domain names can contain alphanumeric characters besides '-' and '.' . However the first and the last character cannot be '-'.
9. The postmaster account is created as an active account.
10. For a clustered domain, the postmaster id is added only for the first domain part of the cluster.

OPTIONS

-V

print version

-v

verbose

-q

quota_in_bytes (sets the quota for postmaster account)

-b

(bounces all mail that doesn't match a user, default)

-e

email_address (forwards all non matching user to this address)

-u

user (sets the uid/gid based on a user in /etc/passwd). Default is user *indimail*.

-d

dir (sets the dir to use for this domain)

-i

uid (sets the uid to use for this domain)

-g

gid (sets the gid to use for this domain)

-a

sets the account to use APOP, default is POP

-O

optimize adding, for bulk adds set this for all except the last one

-f

sets the domain to use **vfilter** mechanism for mail filtering.

-t

Enable ETRN support for the domain.

-T ip_address

Enable AUTOTURN Support from the IP ip_address

-D database

Adds domain information to dbinfo table. This implies use of the -S, -U, -P, -L options additionally.

-S SqlServer

IP address or hostname of **MySQL(1)** server.

-U User

User for accessing MySQL Database.

-P Password

Password for the user to access the MySQL database.

-p MySQL Port

Port to connect to for accessing the database.

-c

Adds a clustered domain. Implies the use of -S, -D, -p, -U, -P options.

[virtual domain]

Name of the new virtual domain

[postmaster password]

The password for the RFC required **postmaster** account. If the password is not supplied on the command line then **vadddomain** will prompt for the password twice.

RETURN VALUE

0 if all steps were successful, non-zero otherwise. If any of the steps fail, a diagnostic message is printed.

vmoddomain[+]

vmoddomain modifies delivery instruction for a domain

SYNOPSYS

vmoddomain [options] domain

DESCRIPTION

Modifies delivery instructions for a virtual domain. All qmail files and system are automatically updated.

OPTIONS

-V

print version

-v

verbose

-f

Sets the domain with VFILTER capability

-h handler

Sets the handler for delivery instruction. The handler can be one of the following values

1. The keyword 'delete'
2. The keyword 'bounce-no-mailbox'
3. Maildir – A full path to a Maildir (e.g. /home/manny/Maildir/)
4. Email Address – Email address for forwarding emails
5. IP Address – An IP address in SMTPROUTE specification (e.g. indimail.org:192.168.2.1:25)

RETURN VALUE

0 if all steps were successful, non-zero otherwise. If any of the steps fail, a diagnostic message is printed.

+ vmoddomain has the setuid bit set and runs with domain's uid

vaddaliasdomain[+]

SYNOPSYS

vaddaliasdomain new_domain old_domain

DESCRIPTION

Take an existing domain and link a new domain to all the accounts on the existing domain.

- Insert into table aliasdomain on the Control Host (if *old_domain* is clustered).
- Create a symbolic link *new_domain* linked to *old_domain* in the domains directory.
- Add entry to qmail *assign* file.
- Add entry to ***virtualdomains*** control file.
- Send SIGHUP to **qmail-send.**
- Add entry to *.aliasdomains* in the domains directory for the domain *old_domain.*

OPTIONS

new_domain

The desired name for the new *alias* domain.

old_domain

The name of an existing *real* domain to which the domain *new_domain* needs to be aliased.

RETURN VALUE

0 if all steps were successful, non-zero otherwise. If any of the steps fail, a diagnostic message is printed.

[+]

1. Only works with cdb authentication module. MySQL module works but only for authentication against the first domain.
2. Has setuid bit set and runs with domain's uid
3. Domain names can contain alphanumeric characters besides '-' and '.' . However the first and the last character cannot be '-'.

vatrn

SYNOPSYS

vatrn [options] email | domain

DESCRIPTION

Enables ATRN domains for On Demand Mail Relay (ODMR). ATRN domains can be added by using the **-t** option to **vadddomain**.

OPTIONS

-s

Displays domains enabled for the user specified by the email address **email** or domain **domain**. If neither email or domain are specified, all atrn domain access are displayed

-d domain

Deletes domain enabled for the user.

-i domain

Adds a new domain for ODMR access by the user.

-u domain

Renames an existing domain for which the user has access to the domain specified in the **-i** option.

-v

Print verbose messages on the screen.

-V

Prints version numbers.

email

The name of an existing user in the indimail database. The encrypted passwd of this user should be used by the ODMR client to authenticate using CRAM-MD5. Users for ODMR can be created by using vadduser (provided domain has already been created using the vadddomain).

domain

Name of an existing ATRN domain. All users having access to download mails using ODMR protocol are shown.

RETURN VALUE

0 if all steps were successful, non-zero otherwise. If any of the steps fail, a diagnostic message is printed.

vdeldomain[+]

SYNOPSYS

vdeldomain [-cTvV] domain_name

DESCRIPTION

Completely removes a virtual domain from the system. Including updating all qmail files and removing all email users and their directories. For a ***clustered domain***, vdeldomain should be run on the host having the *postmaster* account.

1. For *real domain*, it does the following
    - Delete all domains aliased to the real domain being removed (from the file *.aliasdomains*)
    - Remove entry for real domain in qmail *assign* file
    - Recursively remove directories in all filesystems pertaining to the real domain.
    - Remove entries in *dir_control* tables pertaining to the real domain.
    - Remove the Domain directory
    - Remove all entries for domain in tables *indimail* and *indibak*.
    - Remove entries for domain in *rcpthosts, morercpthosts, virtualdomains*
    - Remove entry from table ***dbinfo*** and ***mcdfile*** for a clustered domain.
    - Send SIGHUP to **qmail-send**

2. For *alias domains*, it does the following
    - Delete entry from *aliasdomain* table on the Control Host.
    - Remove entry from *.aliasdomains*
    - Remove entry for alias domain in qmail *assign* file
    - Recursively remove directories in all filesystems pertaining to the real domain.
    - Remove entries in dir_control tables pertaining to the alias domain.
    - Remove the Domain directory (symbolic link)
    - Remove all entries for domain in tables *indimail* and *indibak*.
    - Remove entries for domain in *rcpthosts, morercpthosts, virtualdomains*
    - Send SIGHUP to **qmail-send**

3. For domain with AUTOTURN support, it does the following
    - Recursively remove Maildir for the domain
    - Remove entries for domain in *rcpthosts, morercpthosts, virtualdomains* and *etrnhosts.*
    - Send SIGHUP to **qmail-send**

+Has setuid bit set and runs as root

OPTIONS

-v

Print verbose messages on the screen.

-V

Prints version numbers.

-T

Deletes a domain enabled with AUTOTURN support.

-c

Deletes a domain information from dbinfo table as well.

domain_name

The domain name to remove.

RETURN VALUE

0 if all steps were successful, non-zero otherwise. If any of the steps fail, a diagnostic message is printed.

vdominfo

SYNOPSYS

vdominfo [domain_name]

DESCRIPTION

vdominfo displays domain information for all domains (or for a particular domain given on the command line). The information is given as below

Domain : The domain name (the real domain in case of an alias domain).
uid : uid of the virtualdomain
gid : gid of the virtualdomain
H Id : Hostid for the host where the domain is created.
Ip : IP Address of the host.
Dir : Domain directory for the domain.
Users : No of users in this domain. (Not displayed for an alias domain)
AliasDomains: List of alias domains for this domain.

OPTIONS

[domain_name]
The name of the virtual domain.

-a
Display all fields
-n
Display domain name
-u
Display uid fields.
-g
Display gid fields
-d
Display domain directory
-t
Display user count
-l
Display alias domains.
-p
Display SMTP ports
-V
Display version numbers

RETURN VALUE

0 for success, 1 for any failure.

printdir

SYNOPSYS

printdir domain_name

DESCRIPTION

printdir displays the *Directory Structure Layout's hashed component.* The hashed component is used by vadduser to construct the home directory. Hashing enables faster directory lookups. The information for the directory structure is maintained in the *dir_control* tables.

OPTIONS

domain_name
The name of the virtual domain.

RETURN VALUE

0 for success, 1 for any failure.

vreorg

SYNOPSYS

vreorg -d domain_name [-r] [-R] user_list

DESCRIPTION

vreorg deletes the *Directory Structure layout* and assigns new hashed component for the home directory of the user. The existing files for the user are moved to the new path if necessary. indi*mail*, *indibak*, *valias* entries referring to the earlier path are updated with the new path.

OPTIONS

-d domain_name
The name of the virtual domain.

-r
Reset Dir Structure Layout for domain_name

-R
Do not decrement Dir Structure Layout in the original filesystem.

user_list
Name of a file contain valid users for the virtual domain.

RETURN VALUE

0 for success, 1 for any failure.

vsmtp

SYNOPSYS

vsmtp [options] [host@domain_name -m mta | -s domain_name]

DESCRIPTION

vsmtp sets up artificial routes for **qmail-remote** to be able to deliver mails to the mailstore on host **host** and domain **domain_name.** This route is used by qmail-remote for clustered domains. For all other domains the *control/smtproutes* is consulted. In case, the Control Host is unavailable due to any reason, qmail-remote fall backs to *control/smtproutes* and expects the MDA on the mailstore to re-route the mail to the correct destination.

OPTIONS

-V
: Print Version Numbers

-v
: Sets Verbose option.

-s
: Show smtp ports for a virtual domain or *SMTP Route* between a MTA host and a mailstore having the domain *domain_name*

-d
: Delete *SMTP Route* between a MTA host and a mailstore.

-i port
: Add *SMTP Route* between a MTA host and a mailstore

-u port
: Change *SMTP Route* between a MTA host and a mailstore

-m mta_ipaddress
: The ip address of the MTA which needs a SMTP route to the mailstore.

host@domain
: The name of the mailstore followed by '@' sign and the domain_name.

domain_name
: The name of the virtual domain.

RETURN VALUES

0 for success, 1 for any failure.

vipmap

SYNOPSYS

vipmap [options] ip_address domain_name

DESCRIPTION

vipmap maps an ip address on a mailstore to a virtual domain. In case a host has multiple ip addresses, individual ip addresses can be mapped to individual virtual domains. This can be used instead of the environment variable DEFAULT_DOMAIN or INDIMAIL_DOMAIN. The environment variable INDIMAIL_DOMAIN takes precedence.

OPTIONS

-V
Prints Version Numbers

-v
Sets verbose option**.**

-d ipaddr domain_name
Deletes mapping for ip address ipaddr and domain domain_name

-a ipaddr domain_name
Adds a mapping between IP address ipaddr and domain domain_name

-p
Prints mapping between all IP addresses and virtual domains.

RETURN VALUES

0 for success, 1 for any failure.

vcalias

vcaliasrev.sh

SYNOPSYS

vcalias domain_name
vcaliasrev.sh domain_name

DESCRIPTION

vcaliasrev.sh converts *.qmail* files to IndiMail's optimized *valias* entries. These entries are maintained in the table *valias*. The IndiMail's MDA *vdelivermail* is capable of working with both *.qmail* files as well as valias. Hence, *.qmail* and *valias* can coexist. However care should be taken by the user not have duplicate entries (i.e. Entries maintained as *.qmail* file as well as *valias*.

vcaliasrev.sh does the reverse of **vcalias**. It converts from IndiMail's **valias** entries to *.qmail* files. **vcaliasrev.sh** is a shell script which uses **MySQL** client to query the *valias* table and build the *.qmail* files.

OPTIONS

domain_name

Name of a virtual domain for which the *.qmail* files need to be converted to *valias*.

RETURN VALUES

0 for success, 1 for any failure.

vrenamedomain[+]

SYNOPSYS

vrenamedomain old_domain_name new_domain_name

DESCRIPTION

vrenamedomain renames a real_domain or an alias_domain to a new name. In case of an alias domain, it adds new_domain_name as a new alias domain and deletes the alias domain old_domain_name. In case of a real domain, it does the following

- renames the domain directory.
- renames entries having old_domain_name to new_domain_name in the following tables.
    - indimail, indibak
    - valias
    - dir_control, and filesystem specific dir_control tables
    - lastauth
    - smtp_port
    - userquota
- adds the entry for new_domain_name in qmail's *assign* file.
- deletes the entry for old_domain_name in qmail's *assign* file.
- creates domain directories in the BASE_PATH for A2E, F2K, L2P, Q2S, T2Zsym.
- runs qmail-newu to recreate the *assign cdb* file.
- relinks all alias domains aliased to old_domain_name to new_domain_name.
- creates a file *.domain_rename* in the new_domain_name's domain directory. If this file is present and a user logs in through IMAP4 or POP3, the file .domain is recreated during the login process also. This ensures that when a user logs in, the *.domain* file contains the correct domain name.
- recreates .domain file in user's Maildir to with new_domain_name as the entry.
- deletes the file *.domain_rename* when *.domain* file has been recreated for all the users.

OPTIONS

old_domain_name

Name of an existing domain which is to be renamed.

new_domain_name

Name of a non-existing domain which needs to be given to old_domain_name

RETURN VALUES

0 for success, 1 for any failure.

+ Has setuid bit set and runs as root.

hostsync[+]

SYNOPSYS

hostsync [options] -d domain_name

DESCRIPTION

hostsync is a utility which synchronizes user additions and deletions between the mail host and the control host in a clustered domain setup. It synchronizes the table *indimail* and the *hostcntrl* table on the Control Host. This utility should be setup in **cron(1)** to be run every 30 minutes by the system administrator.

Normally the tables *indimail* and *hostcntrl* should be synchronized. But in case of problems like the control host being down, or the database on the Control Host being down, the tables can get out of sync. **hostsync** corrects all records that have changed on the mailstore, but have not got applied on *hostcntrl* by looking at the pw_uid field of the table *indimail* or *indibak*. Records which are synchronized have the value 1. Value of pw_uid other than 1 indicate the following

pw_uid Value	***Description***
2	User not added to hostcntrl
4	User not deleted from hostcntrl

OPTIONS

-v

Sets verbose option

-d

Name of the clustered virtual domain.

RETURN VALUES

0 for success, 1 for any failure.

+ Only one instance of hostsync can run at a time.

vadduser[+]

SYNOPSYS

vadduser [options] email_address password

DESCRIPTION

Adds a new user to a virtual domain. No additional modifications to the system are required. The account created is inactive. On login (IMAP4 or POP3), the account becomes active immediately. The username should be either alpha-numeric or have the characters '.', '-', '_' . The username, gecos and domain component cannot have the ':' character. The case is changed to lower case before adding to the database. For a clustered domain, the user is also added to the table *hostcntrl* on the Control Host. The *Directory Structure Layout's hashed component* (See Below) is incremented by one.

The user's home dirctory has four components
1. Base Path
2. Directory Prefix
3. Domain Name
4. Hashed Component

The ***Base Path*** is defined by the environment variable BASE_PATH. If this variable is not set, the value is taken from BASE_PATH defined indimail.h. If **vfstab** (-b option) is run periodically, -b option in vadduser can be used to balance optimally user creation across filesystems to distribute the load.

The ***Directory Prefix*** depends on the first character of the username. It can have one of the five values

1. A2E: First character of username is a alphabet including and lying in between 'a' and 'e'
2. F2K: First character of username is a alphabet including and lying in between 'f' and 'k'
3. L2P: First character of username is a alphabet including and lying in between 'l' and 'p'
4. Q2S: First character of username is a alphabet including and lying in between 'q' and 's'
5. T2Zsym: First character of username is a alphabet including and lying in between 't' and 'z' or starts with a non alphabetic character.

[+]
1. User account becomes active immediately.
2. Must be run as root or indimail user. If no domain is specified then must be run as root.
3. Passwd is encrypted using the crypt(2) routine.

The ***Domain Name*** component is derived from the virtual domain (domain component) of the username.

The ***Hashed Component*** is constructed using an adaptive directory structure which is automatically managed by the core indimail api functions "vadduser" and "vdeluser". This structure is known as *Directory Structure Layout*. For sites with 100 users or less, all user directories are stored in the virtual domain directory. For sites that go above 100 users the adaptive directory structure goes into effect. The basic idea is to break up the user Maildir directories across multiple directories and sub directories so that there are never more than 100 user directories in a single directory. You can look at this structure using the ***printdir*** program.

The default directory setup allows for 62 directories in 3 levels and 100 user directories per directory. The total number of user directories is equal to 100 + (62 * 100) + (62 * 62 * 100) + (62 * 62 * 62 * 100) = over 24 million directories. This should be more than sufficient for any site and probably goes beyond the technology of directory structures.

If you are going to be storing large numbers of user directories, make sure you set your file system to have a higher than normal percentage of inodes.
vadduser will automatically create these directories and sub directories as needed and populate each directory with up to 100 user accounts. As soon as a directory reaches 100 users it will create the next directory or sub directory and store the new users directory there.

Over a period of time, due to large no of deletions of users, the *Directory Structure Layout* for all users can be regenerated using the program ***vreorg.***

OPTIONS

-V

Display Version Numbers

-v

Set verbose mode.

-c Comment

Set Comment (Sets the gecos comment field)

-d

Create the directory for the user. If this option is not given, the home directory is not created. It gets created when the user logs in either through IMAP4 or POP3 protocol.

-e passwd

Expects Standard Encryption Password to be given on command line.

-r

Generates Random Password. Need not give password on command line.

-a

Set the account to use APOP instead of the default POP.

-B

Set the base path for the home directories.

-b

Balances users across filesystems listed in ***fstab*** table as they are created. This option should be used if **vfstab** (with -b option) is enabled in cron.

-q [quota in bytes]

Set the hard quota limit for the user. If not supplied then the default system hard quota limit is set. The default limit is either 50M or what ever is set via --enable-hardquota. If set to NOQUOTA then the user will have no quota limit.

-h *Host*

For a clustered domain, this option can be used to create the user on a specific host having hostid as ***Host***.

-m *mdahost*

For a clustered domain, this option can be used to create the user on a specific cluster having Mail Delivery Host as ***mdahost***.

email_address

The new email address of the user. Requires the domain name as well as the user name. For example: user@domain.com. If the domain name is not specified the user is added to the default domain.

password

Set the password for the user. If the password is not supplied on the command line then vadduser will prompt standard in for the password. The password must be typed in twice.

The user is added to the inactive table – ***indibak*** (except for RFC ids ***postmaster*** and ***abuse***) and is treated as an inactive user until the user logs in, upon which the user record is moved to the active table – ***indimail.***

RETURN VALUE

0 if all steps were successful, non-zero otherwise. If any of the steps fail, a diagnostic

message is printed.

vpasswd[+]

SYNOPSYS

vpasswd [options] email_address [password]

DESCRIPTION

Change a virtual domain email users password.

OPTIONS

-V

Display Version Numbers

-v

Set verbose mode.

email_address

username including the domain component

[password]

Set the password on the command line. If the password is not supplied on the command line then user is prompted for the password. The password must be entered correctly twice in order to be accepted.

RETURN VALUE

0 if all steps were successful, non-zero otherwise. If any of the steps fail, a diagnostic message is printed.

[+]

1. Must be run as root or as the indimail user.
2. Passwd is encrypted using the crypt(2) routine

vsetuserquota[+]

SYNOPSYS

vsetuserquota [options] email_address quota_in_bytes

DESCRIPTION

Change a users hard quota limit. --enable-hardquota=y must be set during indimail configuration for this to have any effect. Default configuration sets hard quota on.

OPTIONS

-V
: Display Version Numbers

-v
: Set verbose mode.

email_address
: The email address of the user.

quota_in_bytes
: The hard quota limit in bytes. Abbrivations can be used for kilobytes and mega bytes. For example vsetuserquota user@domain 5m will set the quota to 5 mega bytes. Or vsetuserquota user@domain 2k will set the quota to 2k. vsetuserquota user@domain NOQUOTA will turn off all quota checking for that user.

RETURN VALUE

0 if all steps were successful, non-zero otherwise. If any of the steps fail, a diagnostic message is printed.

[+] Must be run as root or indimail user.

vdeluser[+]

SYNOPSYS

vdeluser [options] email address

DESCRIPTION

Completely removes a virtual domain email account from the system. Takes effect immediately. All files and directories for the user are removed from disk. All forwarding, aliases, entries in database are also removed. The *Directory Structure Layout's Hashed Component* is decremented by one. For a clustered domain, the user will be deleted only if the user lies on the host where **vdeluser** is being run.

OPTIONS

-V

Display Version Numbers

-v

Set verbose mode.

email address

Fully qualified email account. For example: user@domain.com. If no domain is specified then the account is removed from the default virtual domain.

RETURN VALUE

0 if all steps were successful, non-zero otherwise. If any of the steps fail, a diagnostic message is printed.

+

1. Must be run as root.
2. While deleting the home directory, the program does setuid to the domain user for safety

vdeloldusers+

SYNOPSYS

vdeloldusers [options]

DESCRIPTION

vdeloldusers is a utility to delete inactive users or mark users who have not logged in for more than a specified number of days as inactivc. When marking users as inactivc, **vdeloldusers** moves the user records from the active authentication table *indimail* to the inactive authentication table *indibak*. For active users, it deletes unread mails older than age_in_days specified by the -u option. Unread mails are deleted only from the inbox.

OPTIONS

-v

Sets verbose mode while vdeloldusers is operating

-d domain_name

virtual domain for which vdeloldusers is to be run.

-a age_in_days

Delete accounts older than this date. Default is 2 months or 60 days)

-u age_in_days

Delete mails older than this date. Default is 1 months or 30 days. Specifying the age as -1 disables this option.

-t age_in_days

Deletes trash for users who have logged in for the last 'age_in_days' days.

-f

Make vdeloldusers run faster by skipping stat() system call done on Maildir/cur (for ensuring while deleting a user, if a user becomes active). By specifying this option you run the risk of deleting inactive users who may become active while vdeloldusers is running.

-p

Purge entry from Database. The user, home directories and all files pertaining to the user are removed from the system. All records in IndiMail tables are

+ Should be run as root user only.

additionally removed.

-i

Mark the user as Inactive. The user's authentication records are moved from the active table *indimail* to to the inactive table *indibak.* Additionally, INACT record is set for the user in the ***lastauth*** table.

-r

Report only. This option causes vdeloldusers to display the actions it would take but does not actually take the action.

RETURN VALUE

0 if all steps were successful, non-zero otherwise. If any of the steps fail, a diagnostic message is printed.

vgroup

SYNOPSYS

vgroup -a [-h] [-c][q] [-v] [-V] groupAddress [password]
vgroup [-i aliasLine] [-d aliasLine] [-u newaliasLine -o oldaliasLine] [-v] [-V] groupAddress

DESCRIPTION

vgroup implements group functionality in IndiMail. It is a utility to add groups. On an existing group, vgroup can be used to add, modify and delete members.

The first form of usage is used for creating a new group. Subsequently normal IndiMail programs (vuserinfo, vdeluser, vmoduser, vpasswd, vsetuserquota) can be used to operate on a group exactly like on virtual users. In fact, a group in IndiMail is a virtual user.

The second form of usage is used to add, modify, delete aliases to the group. It uses the **valias** to do this. The most comman usage of vgroup is to add forwardings.
i.e. All mails sent to the group will be forwarded to members. In this case address will be of the form '&email_address'.

The implementation of ***groups*** in IndiMail is actually a normal email account with aliases. Hence a group can also be created using **vadduser** in conjunction with **valias** with the restriction that the ***gecos*** starts with the word '**MailGroup**' (see option -c below for more explanation). **vgroup** has been written for user convenience alone.

OPTIONS

Common Options

-v
Sets verbose mode while vgroup is operating

groupAddress
Email address for the group.

Options for adding a New Group.

passwd
Passwd for the user. If not given, user will be prompted.

-a
Add a new group.

-h

Host on which the account needs to be created (valid if group account is being created on a clusted domain).

-c

Comment field (gecos) for the group account. Internally the comment starts with the word 'IndiGroup'. This will be shown if you use the command **vuserinfo** on the group address. It is advisable not to change this by external programs such as vmoduser. IndiMail figures out difference between a normal email account and a group account by checking if gecos field starts with the word 'IndiGroup'.

-q

Sets the Quota for the group. This should be set in case the group will also function as a normal user and receive mails.

-m

Ignore requirement of groupAddress to be a local address on the node where vgroup is being run (option valid for clustered domain only)

Options for an Existing Group

-i aliasLine

Insert aliasLine in the group. See **valias** for more details

-d aliasLine

delete aliasLine from the group. See **valias** for more details

-u newaliasLine -o oldaliasLine

Modify oldaliasLine to newaliasLine alias in the group. See **valias** for more details

RETURN VALUE

0 if all steps were successful, non-zero otherwise. If any of the steps fail, a diagnostic message is printed.

vdelivermail+

SYNOPSYS

vdelivermail [""] [default account]

DESCRIPTION

Called from the virtual domains ***.qmail-default*** file to deliver email to a virtual user. **vdelivermail** is the **MDA** for **IndiMail**. Along with a wrapper **postdel**, **vdelivermail** can be used as the MDA for **postfix** also. **qmail-local** sets the environment variables **EXT** and **HOST** which is used by **vdelivermail** to set the user and the domain component of the email address. For **postfix**, these two environment variables are set by **postdel**. If the username is quoted, the quotes are removed. The user@domain is then searched in the authentication database. If there are any system problems due to which **vdelivermail** is unable to locate the user, it exits with the code 111 and the mail gets deferred. If the domain is a clustered domain, **vdelivermail** attempts to find the host for the user and calls **qmail-remote** to deliver the mail in case the host is different from the current host. Else if QMAIL_EXT is set during compile time, **vdelivermail** attempts to locate the username based on Qmail Extensions. If the user is not found or user is found but is inactive action as per delivery instruction (third argument to **vdelivermail**) is taken. If the user is active, **vdelivermail** checks for any forward delivery (including aliases) and delivers appropriately. While delivering, **vdelivermail** creates the inbox (Maildir format) if the directories do not exist. If the size of the mail is within the quota, or if the size of the mail is less than QUOTA_MAILSIZE (defined in indimail.h) but would bring the user overquota, the mail is delivered and the user's current quota usage updated. Else the mail is bounced back with Over Quota message. Also if /var/indimail/bin/overquota.sh exists, the command is executed with the following arguments

maildir_path Message_size Current_disk_usage Current_mailcount quota

The command overquota.sh can be changed by setting the environment variable OVERQUOTA_CMD. The default behaviour of bouncing mails on overquota can be changed by setting HOLDOVERQUOTA environment variable or by having a file ***holdoverquota*** in the user's homedir. In this case, the mail will be deferred till the queuelifetime gets reached. If HOLDOVERQUOTA is not defined and neither the file holdoverquota is present **vdelivermail** (if the user is already above quota) sets the BOUNCE FLAG for the user. Subsequent deliveries are bounced without any quota checking. Setting of the BOUNCE_FLAG reduces the load on the server when multiple mails are being sent to an overquota user. The BOUNCE FLAG is removed only after the user logs in and clears the mails to reduce the quota usage. Site administrators can customize Over Quota Bounce messages, by setting environment variables QUOTAWARN1, QUOTAWARN2 ... upto QUOTAWARN10. These

+Invoked by qmail-local and run as indimail user.

variables should be set to a percentage quota usage at which warning should be sent. In addition to the EXT and HOST environment variable, **RPLINE** variable is also used by **vdelivermail** to set the value for **Return-Path** in the mail headers.

The quota limit for a user can be set by the administrator either in size, number of mails or combination of both. e.g. 40000,2000S means quota of 40000 Bytes and 2000 mails. Administrator can use either **vsetuserquota** or **vmoduser** programs.

Additionally per day limit on deliveries per user can be set by specifying the environment variable MAILCOUNT_LIMIT. This can be set in the **qmail-send** run file. If the number of deliveries for a user exceeds this number, the mail is bounced back to the sender with over quota message.

vdelivermail defers the mail if stick bit is set on the Maildir to which the mail is being delivered.

vdelivermail looks up the qmail control files ***blackholedsenders*** and ***blackholedpatterns***. If the sender's email address matches an entry in these control files, vdelivermail discards the mail without bouncing the mail to the sender. A line in these files may be of the form *@host*, meaning every address at *host*. These files are also used by **qmail-smtpd** causing SMTP sessions to get blackholed.

blackholedpatterns gives qmail-smtpd the ability to blackhole E-Mails by comparing the sender address with a REGEX pattern in blackholedpatterns.
Example:

*@earthlink.net
!fred@earthlink.net
[0-9][0-9][0-9][0-9][0-9]@[0-9][0-9][0-9].com
answerme@save*
%

blackholedpatterns file with this contents stops all mail from earthlink except from fred@earthlink.net. It also stops all mail with addresses like: 12345@123.com and
answerme@savetrees.com. Further, any E-Mail with a sender address containing a percent sign (%) is rejected.

vdelivermail adds the following headers on delivery of the mail to a Maildir
1. Delivered-To: specifying the address to which the mail has been delivered
2. Return-Path: specifying the sender's email address.
3. X-Filter: specifying if the mail has been filtered through vfilter.
4. Received: specifying the date and time, the mail was received and delivered.

dot-qmail processing

Every virtualdomain get's it's own directory under ~indimail/domains. qmail's user/assign file gets an entry for each domain that points qmail-local deliveries into this directory. Therefore, all normal .qmail file processing works in each virtual domain. .qmail files just need the user name extension to work, i.e. .qmail-joe for user joe. ezmlm uses .qmail files for processing, so it will work under IndiMail.

If no user matches a .qmail file then the .qmail-default file is processed. This file contains the vdelivermail program. This program reads the authentication database (MySQL or vpasswd.cdb) and delivers the mail into the users directory. The last parameter of vdelivermail can be a maildir owned by indimail/indimail so that all default mail reception ends up there. Or it can have an email address, and all default mail is forwarded to this address. Last but not least, the last parameter to vdelivermail can be the text bounce-no-mailbox. This will bounce all non matching emails back to the sender.

In addition to ***.qmail*** files, IndiMail has its dot-qmail processing called ***valias***. ***valias*** can be either the file ***.qmail*** in the user's home directory or an entry made in **MySQL(1)** using the **valias(1)** program.

Mail Alerts

vdelivermail sends alerts in the form of UDP packets to a host and port which can be specified in the following two ways

1. By defining environment variables MAILHOST_ALERT and MAILHOST_PORT.
2. By creating a file mailalert.cfg in the qmail/control directory. The first line of this file should have the line **host *x.x.x.x*** where x.x.x.x is an IP address. The second line of this file should have the line **port *port_num*** where port_num is an integer.

OPTIONS

[""]

Blank double quote for backward compatibility

[default account for delivery]

If the email does match any .qmail-user file and also does not match any virtual domain user, this is the default delivery instructions. This may be one of the four values given below.

- Full path to a indimail user directory
- email address to forward email to
- the string **bounce-no-mailbox** to bounce the email back to the sender.
- Address in SMTPROUTE format to which vdelivermail should use SMTP to

deliver the mail. e.g. indi.com:202.144.76.55:25 where indi.com is the local domain, 202.144.76.55 is the IP address of an SMTP server for indi.com and 25 is the SMTP port.

RETURN VALUE

- 0 if all steps were successful.
- 100 for permanent errors. i.e. if user is over quota or **bounce-no-mailbox** is set and no matching user is found.
- 111 for all temporary error occurs during mail delivery and without the error, the mail would have got delivered

postdel

SYNOPSYS

postdel -u user -d domain -r recipient

DESCRIPTION

Called from the ***postfix's master.cf*** configuration file to deliver email to a virtual user.
postdel sets the environment variables EXT, HOST and RPLINE which is used by **vdelivermail** to set the user and the domain component of the email address. **postdel** forks and calls **vdelivermail** and gets the exit status. The exit status is than mapped to postfix/sendmail compatible exit status. Since **vdelivermail** can handle only one recipient at at time, the variable **default_destination_recipient_limit=1** should be set in main.cf. If default_destination_recipient_limit is set to more than 1, all mails will bounce with usage error. Hopefully this will be corrected in future versions.

OPTIONS

-V
Print version numbers

-v
Sets verbose option

-u
Username to which mail is to be delivered

-d
Recipient's fully qualified email address.

-r
Sender's email address.

RETURN VALUE

- 0 if all steps were successful.
- 67 for permanent errors. i.e. if user is over quota or **bounce-no-mailbox** is set and no matching user is found.
- 75 for all temporary error occurs during mail delivery and without the error, the mail would have got delivered
- 71 for Operating System Errors
- 70 for Software Bugs
- 64 for errors due to misconfiguration

vmoduser

SYNOPSYS

vmoduser[options] email_addr

DESCRIPTION

vmoduser modifies attributes of an email account by modifying fields in the authentication tables *indimail* or *indibak*

OPTIONS

-V

Print Version numbers

-v

Sets verbose mode

-n

Toggles between Inactive and Active state for the user
i.e. Moves the user between *indimail* and *indibak*

-q quota

Changes the quota for the user. Changes the field pw_shell in *indimail* or *indibak*.

-c comment

Changes the gecos (comment) field for the user. Changes the field pw_gecos in *indimail* or *indibak*.

-P clear_text

Changes the password for the user after encrypting the clear text password given on the command line. Changes the field pw_passwd in *indimail* or *indibak*.

-e encrypted_passwd

Changes the encrypted password for the user as given after **'-e'** argument (without performing any encryption). It is expected to supply a crypted password using **crypt(3)** library call. Changes the field pw_passwd in *indimail* or *indibak*.

-l vacation_messsage_file

Sets up autoresponder for the user. It creates .qmail file in user's homedir containing path to the autoresponder program **vacation.** If

vacation_message_file is specified as '-', autoresponder is removed. If it is specified as '+', the text for autoresponder is taken from STDIN. Any other value is taken as a file containing text for the autoresponder.

The following options are bit flags in the gid int field. (pw_gid in *indimail* or *indibak*)

-t
Toggle the bit flags in the gid int field

-u
Set no dialup flag.

-d
Set no password changing flag for the user

-p
Set no POP3 access flag.

-w
Set no web mail access flag.

-i
Set no IMAP4 access flag.

-b
Set bounce mail flag.

-r
Set no external relay flag.

-a
Grant administrator privileges.

-0
-1
-2
-3
Reserved for future use.

-x
Clear all flags

The values of pw_gid corresponding to the above options are as below.

Option	*Value*
NO_PASSWD_CHNG	0x01
NO_POP	0x02
NO_WEBMAIL	0x04
NO_IMAP	0x08
BOUNCE_MAIL	0x10
NO_RELAY	0x20
NO_DIALUP	0x40
QA_ADMIN	0x80
V_USER0	0x100
V_USER1	0x200
V_USER2	0x400
V_USER3	0x800
V_USER4	0x2000
V_USER6	0x8000

RETURN VALUES

0 in case of success and non-zero in case of any failure.

vuserinfo

SYNOPSYS

vuserinfo [options] email_addr

DESCRIPTION

vuserinfo displays following information for a user specified by email *email_addr*.

Name	email_address
passwd	encrypted password.
uid	pw_uid field in indimail or indibak
gid	pw_gid field in indimail or indibak
	Status of the user account
gecos	Description of the user
dir	Home Directory for the user.
quota	Quota in Bytes and Mb for the user.
curr quota	Current Usage in Bytes and Number of mails.
Mail Store IP	IP address of Mail Store where the user lies
Mail Store Id	Host ID for the mailstore host.
Sql Database	Host where the database is located.
Table Name	Name of the table which has the user record (*indimail* or *indibak*
Relay Allow	Indicates POP3/IMAP before SMTP for relay control
Added On	Date when the user was added (with the IP Address)
last auth	Last Authentication Date (with the IP Address)
last POP3	Last Authentication Date on POP3 (with IP Address)
last IMAP	Last Authentication Date on IMAP (with IP Address)
PassChange	Last Password Change Date (with the IP Address)
Inact Date	Last Inactivation Date (with the IP Address)
Activ Date	Last activation date (with the IP Address)

vuserinfo also displays mail folders with usage, forwarding or alias information and filters set for users (if the user have these enabled).

OPTIONS

-a

Display all fields, this is the default

-n

Display the Name only

-p

Display crypted password

-u
Display uid field

-g
Display gid field

-c
Display comment field

-d
Display directory

-q
Display quota field

-l
Display usage times

-m
Display Mail Summary

-V
Print version numbers

RETURN VALUE

0 in case of success, 1 in case of any failure retrieving user information.

valias

SYNOPSYS

valias [options] [email_address | domain_name]

DESCRIPTION

valias adds valias entries for virtual users in the table *valias*. v*alias* entries are equivalent to ***.qmail*** files. **valias** is useful in cases where large number of forwarding/aliases are to be maintained and having numerous physical files on the file system could prove to be cumbersome/inefficient or impossible (i.e. > 32768 on some filesystems).

OPTIONS

-V
Display Version Numbers

-v
Set Verbose mode

-s
Show aliases (domain_name needs to be given)

-d alias_line
Delete alias line corresponding to alias_line

-i alias_line
Insert new alias line

-u old_alias_line -i new_alias_line
Change an existing alias_line

-m
Ignore requirement of email_address to be local (option for clustered domain only)

email_address
Should point to an existing deliverable address on the host where valias is executed.

RETURN VALUE

0 in case of success, 1 in case of any failure.

vmoveuser

SYNOPSYS

vmoveuser email_addr new_homedir

DESCRIPTION

vmoveuser changes the current homedir path to new_homedir, moving all existing files to the new path.

OPTIONS

email_addr
: Fully qualified email address (username + virtual domain component).

new_homedir
: Path to which the homedir of the user needs to be assigned to

RETURN VALUES

0 for success, 1 for any failure.

vrenameuser

SYNOPSYS

vrenameuser [options] old_email_address new_email_address

DESCRIPTION

vrenameuser changes the email address of an existing virtual domain user to a new email address. This can be used to either change the username or to move the user to a new virtual domain lying on the same host. vrenameuser changes only the *pw_name* and *pw_domain* component, and doesn't change attributes like pw_*gecos*, *pw_dir*, etc. But can be used in conjunction with **vmoveuser**, **vmoduser** it can be used to change the homedir, gecos also.

OPTIONS

-V
: Print Version Numbers

-v
: Set verbose mode

old_email_address
: Current email address of the user.

new_email_address
: Desired new email address for the user.

RETURN VALUE

0 if successful, 1 in case of any failure.

vchkpass[+]

systpass

SYNOPSYS

vchkpass alternate_checkpassword_program(s) /bin/false
systpass alternate_checkpassword_program(s) /bin/false

DESCRIPTION

vchkpass and systpass supports authentication for ESMTP AUTH option in qmail-1.03, allowing the LOGIN, PLAIN, and CRAM-MD5 AUTH types. **vchkpass** looks up userid in IndiMail's Virtual Database while **systpass** looks up userids in the systems authentication database (/etc/passwd or /etc/shadow)

As reflected in the **qmail-smtpd(8)** man page, **qmail-smtpd(8)** must be invoked with three arguments: hostname, path of the **vchkpass/systpass** program, and subprogram. If these arguments are missing, **qmail-smtpd(8)** will not advertise availability of AUTH.

vchkpass/systpass reads descriptor 3 through end of file and then closes descriptor 3. There must be at most 512 bytes of data before end of file. If it invokes an alternate_checkpassword_program, it opens a pipe duping the read descriptor as 3 for the alternate_checkpassword_program. The write end of the pipe is duped to descriptor 4 and the data which was earlier read on descriptor 3 is written back on descriptor 4. The alternate program is invoked when vchkpass doesn't support authentication for the supplied login name.

For LOGIN, PLAIN AUTH types, the information supplied on descriptor 3 is a login name terminated by \0 and password terminated by \0. vchkpass encrypts the password using cyrpt(3) routine and compares it against the value present in the passwd database.

For CRAM-MD5 AUTH type, the information supplied on descriptor 3 is a login name terminated by \0, a cram-md5 challenge terminated by \0, and a cram-md5 response terminated by \0. **vchkpass/systpass** encrypts the challenge with keyed MD5 using passwords from passwd database. It's compared with response (3rd parameter) and if they are the same then **vchkpass/systpass** uses execvp to run subprogram with the given arguments. If they differ then it returns -1.

+

1. See http://cr.yp.to/checkpwd.html for more information on the interface.
2. vchkpass supports all of the AUTH types advertised by qmail-smtpd.
3. You may be interested in Krzysztof Dabrowski's cmd5checkpw tool at http://www.elysium.pl/members/brush/cmd5checkpw which supports the three AUTH types supported by vchkpass

If challenge and response differ, **vchkpass/systpass** exits 1. If **vchkpass/systpass** is misused, it may instead exit 2. If there is a temporary problem checking the password, **vchkpass/systpass** exits 111.

hostname is simply used to form the CRAM-MD5 challenge. qmail-smtpd invokes vchkpass, feeding it the username and password, in the case of LOGIN or PLAIN, or the username, challenge, and response, in the case of CRAM-MD5. If the user checks out, **vchkpass/systpass** exits with a status of 0 for the user to be authenticated. Otherwise, subprogram should exit with a non-zero status, or call an alternate checkpassword program. The last subprogram can usually be /usr/bin/false (or /bin/false, depending on your flavor of OS).

MUAs and AUTH types tested with this software are:

- Netscape Communicator 4.76 - LOGIN & PLAIN
- Microsoft Outlook 2000 - LOGIN
- Microsoft Outlook Express 5 - LOGIN
- Eudora 5.0.2 – CRAM-MD5
- sylpheed 0.8.2 – LOGIN, PLAIN and CRAM-MD5

OPTIONS

alternate_checkpassword_program

Path to an alternate checkpassword utility which supports additional authentication advertised by **qmail-smtpd**.

RETURN VALUE

0 if authenticated, 1 on authentication failure or system problems.

clearopensmtp[+]

updaterules

SYNOPSYS

clearopensmtp
updaterules

DESCRIPTION

clearopensmtp removes old IP's added after pop authentication. These IP's are maintained in the table *relay* by default. IndiMail can be configured to allow POP3/IMAP users IP's to be added to the list of IP's which are allowed to relay through the smtp server. This is done with the ***--enable-roaming-users=y*** option while configuring IndiMail before compilation. With this option, users IP's who authenticate via POP3/IMAP are added to the list of IP's which can relay through the smtp server. **qmail-smtpd(8)** must be run with **tcpserver(1)** -x option. If the host is a clustered domain, **clearopensmtp** uses the DBINFO structure to cycle through all the mailstores and clear the *relay* tables.

updaterules rebuilds the **cdb (tcp.smtp.cdb)** file with the list of static IP's stored in *tcprules (tcp.smtp)* file and the list of IP's in open-smtp or *relay* table. This rebuilding is needed only if IndiMail's **qmail-smtpd(8)** has not been installed, and one is running some other version of smtpd daemon. **updaterules** may also be needed if one is running a version of smtpd on some foreign hosts (e.g. Relay servers) which do not have access to IndiMail's authentication databases. IndiMail's **qmail-smtpd(8)** gets the authenticated users at run time from the table *relay* and hence does not require the IP's in the t*cp.smtp.cdb* file.

Each authenticated pop users IP is added with a time stamp. Every time clearopensmtp is run, this list is checked for time stamps which are older than the ***--enable-relay-clear-minutes*** option. The default is 30 minutes. Any IP with a time stamp older than this number are removed from the list.

OPTIONS

None

+

1. These programs are required only when IndiMail is configured with ***--enable-roaming-users=y***. It should be run from roots crontab every hour.
2. Not using IndiMail's qmail-smtpd can drastically impact performance of POP3/IMAP4 servers as they will then require to frequently update *tcp.smtp.cdb* when a user logs in.

RETURN VALUE

0 if all steps were successful, non-zero otherwise.

vbulletin[+]

SYNOPSYS

vbulletin [-f filename] [-e exclude email addresses file] [-v] [-n] [-c] [-h] [-s] [-S subscriber_list_file] [virtual domain ...]

DESCRIPTION

vbulletin implements instant bulletin for IndiMail. It does this by copying an email file in the user's inbox (Maildir). The various options given on the command line decide how this copying is to be done.

OPTIONS

[-v]
Verbose mode. Prints out each email address it is sending to.

[-f filename]
File containing the email message to be posted.

[-e exclude email addresses file]
File containing a list of email addresses to exclude from posting.

[-n]
Don't actually mail it. using -v and -n can be used to list out all virtual domain email accounts.

[-c]
Default, copy message to users directory.

[-h]
Make a hard link from email file to virtual users directory. Email file must be on the same physical device as the virtual users directories. This will save disk space.

[-s]
Make a soft link from the email file to the virtual users directory. This will save on disk space but will not remove the file when all users read it. If the original file is deleted, users will not be able to read the message.

-S subscriber_list
Take the list of subscriber list from a file and populates the bulkmail table. The

+ Must be run as root.

IMAP4/POP3 daemon query this table when a user logs in and creates a link in the user's inbox (Maildir) to the email_file. This causes instant delivery of the bulletin without using any disk space. For a clustered domain, vbulletin can be safely run on all the hosts giving the same subscriber_list file. vbulletin takes care of identifying the subscribers listed in subscriber_list file present on the host and those which are present on some other host.

-a

Sends an instant bulletin to the entire domain. The file is copied to the bulk mail directory with ,all appended.

[virtual domain ...]

List of domains to send the message to. If this is not supplied then the message is sent to all virtual domains.

RETURN VALUE

0 if all steps were successful, non-zero otherwise. If any of the steps fail, a diagnostic message is printed.

vacation

SYNOPSYS

vacation [email_addr] [vacation_mesg_file]

DESCRIPTION

vacation implements auto reply for virtual domain users. Without any arguments, the only usage is in *.qmail* files as a valias with *valias_line* as
'| /var/indimail/bin/vacation'

When arguments *email_addr* and *vacation_mesg_file*, auto reply is setup for the user specified by *email_addr*

OPTIONS

email_addr
: Fully qualified email address for a virtual domain user.

vacation_mesg_file
: A file containing the text for auto reply. If the filename is '-' (hyphen), any existing auto-reply is removed for the user. If the filename is '+' (plus sign), the text for auto-reply is taken from STDIN.

RETURN VALUE

0 if successful, non-zero for any failure. Diagnostics are printed on STDOUT.

indiversion

versioninfo

SYNOPSYS

indiversion
versioninfo

DESCRIPTION

indiversion and **versioninfo** display the version numbers and **rcs(1)** information for all modules and source files comprising the IndiMail system. **indiversion** also displays the configuration options set by the **configure(1)** program before compiling IndiMail. It is meant for use by ***Experts from IndiMail*** only.

OPTIONS

None

RETURN VALUE

Return value does not have any importance.

indisrvr

SYNOPSYS

indisrvr -i IP_Address -p port -b backlog [-n certfile]

DESCRIPTION

indisrvr is the administration server for administering IndiMail's clustered domains. It has a simple mechanism by which administrators can connect to this server and issue commands to execute any of the IndiMail programs given above. These commands are issued by the admin client **adminclient** program. e.g. To run **vadduser** on host phoenix2, run **adminclient** on any host on the network. **indisrvr** implements simple authentication using **crypt(3)**. The userid and passwd are maintained in the table ***mgmtaccess***. A separate password maintenance tool **mgmtpass** maintains the passwords in this table. On successful validation of password, **indisrvr** executes the specified IndiMail administration program on the localhost. For security reasons, the list of these administration programs are hard-coded in the **indisrvr** executable. On SIGUSR2, verbose option gets toggled (see -v option below).

e.g. The following command runs the command vadduser with arguments test@indi.com password.
adminclient -h phoenix2 -p 4000 -u admin -P adminpass -c "vadduser test@indi.com password"

OPTIONS

-v verbose
Sets verbose option.

-i IP_Address
The IP address on which indisrvr needs to bind. Useful if a host has multiple IP addresses.

-p port
A TCP/IP port or service in /etc/services on which indisrvr needs to bind.

-b backlog
Max backlog to be maintained after which the client should get connection refused

-n certfile
A PEM certificate causes indisrvr to negotiate TLS encryption with clients.

RETURN VALUE

Always returns non-zero status, which means the server died due to some reasons. Most likely of a bug. Will also return on SIGTERM with exit status=0.

adminclient

SYNOPSYS

adminclient -h adminHost -p port -u adminUser -P adminPassword [-n certfile]
-c Command

DESCRIPTION

indisrvr is the administration client for administering IndiMail's clustered domains. It has a simple mechanism by which administrators can connect to the administration server - **indisrvr** and issue commands to execute any of the IndiMail programs given above. On successful validation of password, **adminclient** executes the specified IndiMail administration program on the remote host running **indisrvr**. For security reasons, the list of these administration programs are hard-coded in the **adminclient** and **indisrvr** executable. **adminclient** uses environment variable ADMIN_TIMEOUT to timeout connections to **indisrvr**.

e.g. The following command runs the command vadduser with arguments test@indi.com password.

adminclient -h phoenix2 -p 4000 -u admin -p adminpass -c "vadduser test@indi.com password"

OPTIONS

-v
Sets Verbose option.

-h adminHost
The IP address on which command needs to run.

-p adminPort
A TCP/IP port or service in /etc/services on which indisrvr is running on adminHost.

-u adminUser
The administration user (maintained in mgmtaccess) for which indisrvr will validate the password.

-P adminPass
The password for the user specified by the -u option.

-n certificate
A client certificate which should be provided in case TLS encryption is needed

-c Command

The command that needs to executed on the remote host adminHost. Command should be enclosed in quotes for commands requiring multiple arguments on the command line.

The -h, -p, -u, -P, -v are optional. If not given, the environment variables ADMIN_HOST, ADMIN_PORT, ADMIN_USER, ADMIN_PASS will be used for these variables.

RETURN VALUE

Returns the return status of the remote command executed. All output printed by the remote command on stdout/stderr are printed on stdout of adminclient.

mgmtpass

SYNOPSYS

mgmtpass -u user [-a passwd] [-p passwd] [-i]

DESCRIPTION

mgmtpass is a passwd program for setting passwords in the table **mgmtaccess.** This table is used the the administration server **indisrvr**. For users other than the superuser, the old password will be prompted and only after successful authentication, will the password be changed.

OPTIONS

-u user

The user for which the password needs to be changed.

-a passwd

This option creates a user. The user has to supply the password for the user '**admin**' to access this.

-p passwd

The superuser can use this option to set/reset the password for an existing user.

-i

Gives information like last accessed date, last update date, no of unsuccessful attempts and whether the user is enabled/disabled. Only the superuser can use this option.

NOTE:
The userid '***admin***' has superuser privileges (like creating other users, resetting passwords, etc). The user 'admin' is created if it does not exist. The user will be prompted for setting password for 'admin' user.

RETURN VALUE

0 for success and 1 for failure in which case all diagnostics are printed on stdout.

vpriv

SYNOPSYS

vpriv [options] user CommandLineSwitches

DESCRIPTION

vpriv sets privileges for users to run IndiMail programs with a set of command line options. The users can be created with the **mgmtpass** program. This allows access control on programs run by **adminclient**.

OPTIONS

-s

Shows all allowed programs and command line switches enabled for the user '***user***'.

-d program

Removes all access to program '***program***' for user '***user***'

-i program

Adds the program '***program***' in the access control list for the given user. Further access is given by the ***CommandLineSwitches*** argument.

-m program

Modifies the command line switches for the program '***program***'

-D

Removes all access to all programs for the user '***user***'

-a

Adds all IndiMail programs to the access control list of the user '***user***' with all possible command line arguments.

NOTE:
Access control list applies only to programs run through the **adminclient** program. The access control list is checked by the **indisrvr** program after successful authentication.

RETURN VALUE

0 for success and 1 for failure in which case all diagnostics are printed on stdout.

proxyimap

proxypop3

vproxy

SYNOPSYS

proxyimap path_of_imapd Maildir
proxypop3 path_of_pop3d Maildir
vproxy host port [login_sequence]

DESCRIPTION

proxyimap and **proxypop3** act as proxy for IMAP4 and POP3 protocol in a clustered domain where users homedir lie on multiple hosts. Proxyimap and proxypop3 correctly locates the user's mailstore and make a IMAP4 or POP3 connection to the user's mailstore host. This facilitates to have a single IP address or DNS entry for all hosts comprising a clustered domain. The proxy works by passing all data from the file descriptor 0 to the socket connection (IMAP4 or POP3). All data which comes from the socket is is passed to file descriptor 1. The recommended method for using **proxypop3** and **proxyimap** is through **tcpserver(1)**. It is recommended to use a cdb files (**-x** option in tcpserver) ***tcp.imap*** for **proxyimap** and ***tcp.pop3*** for **proxypop3**. For enabling unrestricted IMAP from a host, set environment variable POP3CLIENT and IMAPCLIENT in cdb file used by the respective tcpservers (compiled by tcprules). e.g. To allow unrestricted IMAP/POP3 access (regardless of the ***mails attribute*** in the above table) from the host with IP w.x.y.z, use tcprules to compile the following.

tcp.imap
w.x.y.z:allow,IMAPCLIENT=""

tcp.pop3
w.x.y.z:allow,POP3CLIENT=""

Following environment variables are needed by these two proxies to start up

- TCPREMOTEIP
  IP address of remote host.

- TCPLOCALPORT
  Local TCP/IP Port on which proxyimap, proxypop3 is running.

- DESTPORT
  Of the form *protocol:port* where *protocol* is either imap or pop3. Port is the

TCP/IP port on the remote host offering the imap or pop3 protocol.

Currently proxyimap and proxypop3 allow 5 bad login attempts, after which the connection is terminated.

There are two method for automatic mailbox creation

1. Using **indisrvr**

   The following extra environment variables are required.

   - ADMIN_HOST - Host on which indisrvr is running
   - ADMIN_PORT - Port on which indisrvr is listening
   - ADMIN_USER - Userid for indisrvr
   - ADMIN_PASS - Password for indisrvr

It is also possible to set data timeout for the proxies through the environment variable DATA_TIMEOUT. The two proxies will terminate if it finds no data coming from either the client or the actual imap/pop3 server within the time in seconds specified by DATA_TIMEOUT.

vproxy is a generic proxy which passes data from file descriptor 0 to a TCP/IP connection and data from the TCP/IP connection to file descriptor 1. The TCP/IP connection is specified by *host* and *port* parameters. ***login_sequence*** specifies initial data (if needed), that needs to be supplied to the application listening on port *port* on host *host*.

OPTIONS

path_of_imapd
Specifies the full path of IndiMail's imapd executable

path_of_pop3d
Specifies the full path of IndiMail's pop3d executable

host
Host Name (Host Name or IP address for which the proxy service is needed.

port
TCP/IP Port on *host* for which proxy services is needed.

login_sequence
Initial data needed by some servers to start up properly. e.g. POP3 service will require USER username and PASS passwd sequence to login.

RETURN VALUE

0 for success and 1 for failure in which case all diagnostics are printed on stdout.

vhostid

SYNOPSYS

vhostid [options] [*HostId*]

DESCRIPTION

vhostid displays the hostid to IP address mapping used by IndiMail for clustered domains. Each host in a clustered setup is assigned a unique hostid with the help of vhostid. Subsequently, **vhostid** can be used to display, insert, update or delete these mappings. Any host added/deleted/modified is accomplished by using this program.

OPTIONS

-V
: Print Version Numbers

-v
: Sets Verbose Option

-s
: Display hostid Mappings for all hosts (if hostid is not given on command line. Else displays the hostid to IP Address mapping only for the host with id ***HostId***.

-d *HostId*
: Delete IP address Mapping for ***HostId***

-i *Ipaddr*
: Maps IP address ***Ipaddr*** to hostid **HostId**

-u *Ipaddr*
: (Map HostId to New_Ipaddr)
Changes IP address for hostid ***Hostid*** to IP address ***Ipaddr***

RETURN VALUE

0 if successful, 1 for any failure.

dbinfo

SYNOPSYS

dbinfo -i -S MySQL_server -p port -D database -U user -P password [-c] -d domain -m mdahost filename
dbinfo -u -S MySQL_server [-p port -D database -U user -P password] [-c] -d domain -m mdahost filename
dbinfo -r -d domain -m mdahost filename
dbinfo -s [-m mdahost] filename
dbinfo -e

DESCRIPTION

dbinfo is a utility for administering the ***dbinfo*** structure maintained in **MySQL(1).** The above five different forms of usage allows you to add, modify, delete and display the ***dbinfo*** structure. With the **'-e'** option, it uses the EDITOR environment variable to edit the ***Mail Cluster Definition*** file. For reliability, this structure is maintained in MySQL as well as a file defined by the ***MCDFILE*** environment variable (defaults to /var/indimail/control/sql). On updation of the dbinfo structure, the entries in the file is synchronized with the entries in MySQL table ***dbinfo***. In addition to the MySQL table ***dbinfo***, **dbinfo** also uses the qmail assign file to find domains not listed in the dbinfo table. These domains are assumed to be non-clustered domains. The dbinfo structure for this domains is created automatically using the environment variables MYSQL_HOST, MYSQL_USER, MYSQL_PASSWD, MYSQL_DATABASE, MYSQL_VPORT (If MYSQL_HOST is not defined, the control file mysql_host is used).

The ***dbinfo*** structure holds the following parameters

- MySQL Server IP addresses
- MySQL TCP/IP port
- MySQL database which will contain IndiMail's database.
- Username to access the MySQL database
- Password for the Username
- Domain Name which will be served by the MySQL database
- Mail Store IP address.
- Optional Cluster flag indicating that the mail setup is part of a mail cluster.
- Filename pointing to the path of the Mail Cluster Definition File.

The option **'-s'** in **dbinfo** displays the status of all MySQL servers used by IndiMail listed in the *MCD* file. If MCD filename is not given on the command line, the environment variable MCDFILE is used. If environment variable is not set, the static definition of MCDFILE in indimail.h is used. The information displayed is as follows.

Domain	satyam.net.in	Non clustered
sqlserver[003]	202.144.76.7	

mda host	202.144.76.48	
TCP/IP Port	3306	
database	indimail	
user	indimail	
password	Ssh-1.-5-	
fd	4	
MySQL Stat	Uptime: 6105 Threads: 2	Questions: 15 Slow queries: 0
	Opens: 7 Flush tables: 1	Open tables: 1 Queries per second avg: 0.002

OPTIONS

options:

-v

Sets Verbose operation

-i

Insert Dbinfo Entry. All dbinfo parameters are mandatory for this operation

-u

Update dbinfo Entry. With this option the MySQL parameters pertaining to a mdahost serving a domain can be modified.

-r

Remove dbinfo Entry

-s

Select dbinfo Entries

-d

Domain Name

-c

Sets the Domain for Clustered operation.

-S

MySQL Server IP

-p

MySQL TCP/IP Port

-D

MySQL Database Name

-U

MySQL User Name

-P

MySQL Password

-m

Mail Store IP

-e

Edit Dbinfo using vi

filename

Name of a file containing MySQL, MDA configuration for IndiMail

RETURN VALUE

0 for success, 1 for any failure.

ipchange

SYNOPSYS

ipchange [options] table_name

DESCRIPTION

ipchange is meant for modifying IP address records in the **IndiMail** tables when the IP address of a host running various IndiMail services is changed. Following tables get impacted when IP address of a host gets changed.

Table	***Column***
fstab	host
host_table	ipaddr
dbinfo	server, mdahost
ip_alias_map	ipaddr
smtp_port	host, src_host

e.g. You can use the following command to set things right in the table dbinfo
ipchange -o 202.144.76.7 -n 202.144.76.8 -c server -m dbinfo
ipchange -o 202.144.76.7 -n 202.144.76.8 -c mdahost -m dbinfo
ipchange -o 202.144.76.7 -n 202.144.76.8 -c ipaddr ip_alias_map

The control file ***localiphost*** on the host whose IP address is changed, gets impacted. The control file ***smtproutes*** on all relay servers acting as a relay for the host gets impacted.

Please Note that **ipchange** changes only the logical IP address in the IndiMail's database. For the actual/physical IP address change, you will have to use the appropriate OS utility (i.e. ifconfig).

OPTIONS

-v

Sets verbose operation

-o

Existing IP address.

-n

New IP address to be set on the host.

-c

Column name of the IP address record.

-m

Modifies the table on hostcntrl.

table_name

The name of the table.

RETURN VALUE

0 for success, 1 for any failure.

hostcntrl

SYNOPSYS

hostcntrl [options] EmailID

DESCRIPTION

hostcntrl is meant for listing, adding, deleting, modifying user records in the ***hostcntrl*** table for a clustered setup. This is especially useful when you may move a user from one host to another host in a clustered setup.

OPTIONS

-v

Sets verbose operation

-s

Shows records pertaining to EmailID in the ***hostcntrl*** table.

-i

Inserts records in the hostcntrl database. Additional argument hostid has to be given to the -i option.

-m

Modifies the hostid field for the given EmailID.

-d

Deletes the record for the given EmailID.

EmailID

The Email ID for the user.

RETURN VALUE

0 for success, 1 for any failure.

InLookup

SYNOPSYS

InLookup [-f infifo -i instance -v]

DESCRIPTION

InLookup is a connection pooling server to serve requests for *inquery()* function. It is implemented over two fifos. One fixed fifo for reading the query and reading the path of a randomly generated fifo. The randomly generated fifo is used for writing the result of the query back.

It creates a read fifo determined by the environment variable INFIFO. If INFIFO is not defined, the default fifo used is */var/indimail/control/inquery/infifo.* It then goes into an infinite loop reading this fifo.

The write fifo is generated and created by the client through the inquery() function and the path of this fifo is passed along with the query.

InLookup helps in optimizing connection to **MySQL(1),** by keeping the connections persistent. The clients send requests for **MySQL(1)** queries to InLookup through the function ***inquery()*** using a fifo. Clients which are currently using inquery are qmail-smtpd(1), proxyimap, proxypop3, vchkpass and authindi(1). **InLookup** uses the ***MCD*** (Mail Cluster Definition File) defined by the **MCDFILE** environment variable to load all databases. The MCDFILE defaults to ***/var/indimail/control/mcdinfo***. On start up **InLookup** can be setup into debug mode by giving '**-v**' argument on the command line. If debugging is desired to be changed during execution of the server, set the environment variable DEBUG instead. Debugging can then be turned off/on anytime by giving the SIGUSR2 signal to the pid of the **InLookup** or pid of the parent **InLookup** (in case of multiple instance).

The program **inquerytest** simulates all the queries which **InLookup** supports and can be used as a test/diagnostic tool for submitting queries to **InLookup**.

OPTIONS

-v

Sets verbose option.

[-f infifo]

Name of a fifo to read query requests for back end database like **MySQL(1).** This overrides the environment variable INFIFO.

[-i instance]

Start multiple instances of InLookup (as given by 'instance') for load balancing purpose. The default is one instance. If you find that your queries are taking a long time, you can increase this value.

SIGNALS

InLookup reacts to a set of signals.

- SIGHUP
  This lets **InLookup** clear all cache.
- SIGINT
  This lets **InLookup** to close all current connections to MySQL(1) and reread the ***MCD*** Control file.
- SIGUSR1
  This causes InLookup to dump all current statistics on stdout
- SIGUSR2
  This causes InLookup to toggle the debugging flag. If debugging is off, it is set to on and if it is on it is set to off.
- SIGTERM
  This causes InLookup to terminate.

RETURN VALUE

Returns on SIGTERM with exit status = 1.

inquerytest

SYNOPSYS

inquerytest query_type infifo email [ip_address]

DESCRIPTION

inquerytest is a utility to test the inquery() function and its interaction with *InLookup*. This function is meant to be used by qualified IndiMail administrators.

OPTIONS

query_type

Type of query to be tested as given below

1 – User Query
This displays the status of the user, whether the id has all service enabled or is absent, inactive or over quota

2 – Relay Query
Tells whether the user has authenticated and the entry is present in the *relay* table and has not expired (used for POP before SMTP).

3 – Password Query
Displays the passwd structure for the email address.

4 – Host Query
Displays the SMTPROUTE for the address. i.e. To which host should the mails for this address be delivered.

5 – Alias Query
Displays all the aliases, forwarding, vacation entries for the address.

7 – Local Password Query
Similar to (3) except that the email address is checked only on the local MDA..

infifo

Path of the fifo to be used for communication with *InLookup*. A null string (“”) makes inquerytest to use INFIFO environment variable if defined or the default INFIFO file and use an existing running InLookup. If a fifo path is given, inquerytest will invoke a new InLookup on the given fifo, run the test, terminate InLookup and ***delete*** the fifo after the test. Hence, do not give a path to a fifo on which an existing InLookup is running.

email

Fully qualified email address.

ip_address

IP address (TCPREMOTEIP) recorded in relay table after successful

authentication through IMAP/POP3 for the email address *email*. This option is valid for the Host Query type.

RETURN VALUE

0 for success, 1 for any failure. Prints diagnostics on stdout and failures on stderr.

vfstab

SYNOPSYS

vfstab -d | -i | -l | -u | -o | -n user_quota -q size_quota -m mdaHost Filesystem
vfstab -s [-m mdaHost]
vfstab -b

DESCRIPTION

vfstab is a utility for maintaining the ***FSTAB*** table. The ***FSTAB*** table maintains the filesystems which can be used for serving the user mailboxes. Normally when creating an user account (**vadduser**), the environment variable BASE_PATH is used. This however is not optimal when having multiple filesystems. In such case, the administrator can use the -b option in vadduser to distribute the users across multiple filesystems, taking into account the load of the different filesystems. The load factor is computed by running **vfstab** with -b option periodically through **cron(1)**.

OPTIONS

-V
Print version numbers

-v
Sets verbose option

-d
Delete local/remote file system

-i
Insert local/remote file system

-u
Update local/remote file system

-o
Make local/remote file system offline/online

-l
Add local file system. If -n and -q options are not given, the values user_quota and size_quota are automatically calculated from the free disk space available and by assuming an average size of the mailbox to be the value of the environment variable AVG_USER_QUOTA

-n
Max number of users to be added on this file system.

-q

Max size of file system

-m

Mail Delivery Host. The host on which the user's mailbox lie.

-s

Show filesystems configured in ***FSTAB*** table. If -m option is given, displays the file system only for the specified Mail Delivery Host.

-b

Balance filesystems by computing load.

Filesystem

Name of a valid file system.

RETURN VALUE

Returns 0 for success, 1 for any failure.

vcfilter

SYNOPSYS

vcfilter [options] emailid [mailing_list1..mailing_listn]
vcfilter -m emailid [options] [mailing_list1..mailing_listn]

DESCRIPTION

vcfilter is a utility for creating, modifying, deleting IndiMail's powerful filter called as ***vfilter.*** There are two options; one for administration of filters and another for administration of mailing lists subscribed by the user. For mailing lists special action needs to be performed by **vfilter**, so that valid mails do not get rejected. Currently, **vcfilter** allows maximum of 255 filters to be created.

The first form of usage is primarily for creating filters (though some amount of mailing list updation is also possible). The second form of usage is for mailing list updation only.

OPTIONS (for filters)

-v
Sets verbose option

-s
Show Filters

-i
add filter

-d filter_no
delete filter(s)

-u filter_no
update filter

-t filter_name
Textual name for the filter

-h

header value. A number denoting the following

-1 – If comparison (**-c** option) is 5 or 6	
0 – Return-Path	1 – From
2 – Subject	3 – To
4 – Cc	5 – Bcc
6 – Reply-To	7 – Sender
8 – List-Id	9 – Mailing-List
10 – X-Sender	11 – X-Mailing-List
12 – X-ML-Name	13 – X-List
14 – X-Loop	15 – X-BeenThere
16 – X-Sequence	17 – X-Mailer
18 – Precedence	

-c

Comparison with the headers to make. This could be one of the following number.

0 – Equals	1 – Contains
2 – Does not contain	3 – Starts with
4 – Ends with	5 – Sender not in address book
6 – My ID not in To, Cc, Bcc	7 – Numerical or Logical Expression
8 - RegExp	
Note: Mailing List can be specified for Comparison 5 or 6	

-k keyword

A keyword or string which will be used in comparison with the header value (depending on -c option). The string should be a NULL "" string if -c option is either 5 or 6. The keyword can also be any Regular Expression (i.e. *, ? [], etc) if comparison equals 7.

-f Folder

Folder to which the mail should be delivered if a filter match occurs. This can be the Trash folder in case the mail needs to be rejected. It can be /NoDeliver if the mail needs to be discarded completely.

-b bounce_action

One of the four numbers

0. Do not bounce
1. Bounce mail to sender with the following custom message
   "Hi. This is the IndiMail MDA for <domain_name>.
   I'm afraid I cannot accept your message as my SPAM filter has decided

to reject this email.
Please refrain from sending such mail in future."

2. Forward mail to address following the number. An ampersand '&' sign must separate the number and the address. i.e. 3&editor@indi.comwill forward all rejected mails to the mailbox editor@indi.com
3. Same as above but bounce as well.

mailing_list1...mailing_listn

List of mailing lists which should be recognized by vfilter (if the filter comparison (-c option) is 5 or 6. This could also be a NULL "" string in which case, vfilter will intelligently try to figure out mails from mailing lists.

OPTIONS (for mailing lists)

-v

Sets verbose option

-m

Show Mailing list which should be allowed in case filter rule rejects the mail.

-i

add mailing lists

-D

delete mailing lists

-U old_mailing_list_name new_mailing_list_name

Change name of an existing mailing list to another name.

-o option (used only for comparison 5 or 6)

Change behavior when figuring out mailing lists for comparison 5 or 6.
Following are the valid options
0 – Do not consider mailing lists in filtering
1 – Consider mailing lists while filtering intelligently
2 – Consider only mailing lists specified in the filter when filtering.

mailing_list1...mailing_listn

List of mailing lists to be added or deleted.

RETURN VALUE

Returns 0 for success, 1 for any failure.

vfilter

SYNOPSYS

vfilter "" default_account

DESCRIPTION

vfilter is a *MDA* which implements IndiMail's powerful vfilter mechanism. These filters can be set up by the **vcfilter** program. It runs through all filters one by one and takes action the moment the first match is found. Once a match is found, **vfilter** ignores other filters (which could also result in a match). Hence care should be taken by the user to have filters in the proper order in case where multiple matches are possible.

vfilter can be enabled by replacing **vdelivermail** in the ***.qmail-default*** file. For optimization, **vfilter** skips filters if the file ***.vfilter*** is not found in the user's Maildir patch. This file is created by **vcfilter** when the first filter is added and removed when the last filter is deleted. In case while processing, any fatal error occurs, **vfilter** passes the mail to **vdelivermail** directly (or any other mail delivery agent defined by the MDA environment variable) without any further processing. In case of temporary errors, the mail remains in the queue and retried later and hence again passes through the filter. **vfilter** can perform various actions on a filter match. These actions are

1. Delivering to any one specific mailbox (Maildir format – if this is different from Inbox, the environment variable **MAILDIRFOLDER** is set, causing **vdelivermail** to deliver mail to a different folder).
2. Bouncing the mail back to sender
3. Forwarding the mail to another address
4. Running the email through another program

On a successful match, the mail is removed from the queue. If mail passes through the filter without any match, mail remains in the queue and is handed over to **vdelivermail** (or the mda defined by the MDA environment variable) for further processing.

These actions make **vfilter** a powerful mechanism to organize your emails, spam control, automation based on email, taking specific action on receipt of specific emails, etc. In fact a complex email based system can be developed around **vfilter** mechanism. However, due to complexity and current limitations, forwarding are not checked for errors.
vfilter can also be configured to run global filters. There are two global filters

1. Pre Filter

This can be set by setting filters for the special user **prefilt@virtual_domain** by using the program **vcfilter**. All filters defined for this user will be applied for all users before applying the user defined filters. Pre-filters can be turned off for specific users by having a file named ***.noglobal*** in the user's Maildir.

2. Post Filter

This can be set by setting filters for the special user **postfilt@virtual domain** using the **vfilter** program. These filters will be applied for all users after applying the user defined filters. Post-filters can be turned off for specific users by having a file named ***.nopostfilt*** in the user's Maildir.

vfilter can also be run in test mode by specifying emailid of the user on the command line. The only other arguments permitted in test mode is the -v option. If the -v option is specified, all actions performed are shown on screen. This option can be used to test so that important mails do not get lost due to misconfiguration of filters.

vfilter uses a sophisticated mechanism called ***Email Processing System*** (**EPS**) to parse the email (headers, mime and body). **EPS** is a set of API calls which allow to understand the contents of an email message. The fact that email messages are rather complex, means that **EPS** cannot jump around the email randomly like a monkey and be expected to understand its content. **vfilter** runs the email through **EPS** line by line to understand specific things like MIME attachments.

OPTIONS

[""]

Blank double quote for backward compatibility (with **vdelivermail**)

[default account for delivery]

If the email does match any .qmail-user file and also does not match any virtual domain user, this is the default delivery instructions. There are three values this may contain.

- Full path to a indimail user directory
- email address to forward email too
- the string **bounce-no-mailbox** to bounce the email back to the sender.

RETURN VALUE

- 0 if all steps were successful.
- 100 for permanent errors. i.e. if user is over quota or **bounce-no-mailbox** is set and no matching user is found.
- 111 for all temporary error occurs during mail delivery and without the error, the

mail would have got delivered

vmoddomlimits

SYNOPSYS

vmoddomlimits [options] domain

OPTIONS

-v (display the indimail version number)

-S (show current settings)

-D (delete limits for this domain, i.e. switch to default limits)

-Q quota (set domain quota)

-M count (set domain max msg count)

-q quota (set default user quota)

-m count (set default user max msg count)

-P count (set max amount of pop accounts)

-A count (set max amount of aliases)

-F count (set max amount of forwards)

-R count (set max amount of autoresponders)

-L count (set max amount of mailing lists)

the following options are bit flags in the gid int field

-g "flags" (set flags, see below)

gid flags:

u (set no dialup flag)

d (set no password changing flag)

p (set no pop access flag)

s (set no smtp access flag)

w (set no web mail access flag)

i (set no imap access flag)

r (set no external relay flag)

x (clear all flags)

the following options are bit flags for non postmaster admins

-p "perm flags" (set pop account flags)

-a "perm flags" (set alias flags)

-f "perm flags" (set forward flags)

-r "perm flags" (set autoresponder flags)

-l "perm flags" (set mailing list flags)

-u "perm flags" (set mailing list users flags)

-o "perm flags" (set mailing list moderators flags)

-x "perm flags" (set quota flags)

-z "perm flags" (set default quota flags)

perm flags:

a (set deny all flag)

c (set deny create flag)

m (set deny modify flag)

d (set deny delete flag)

RETURN VALUE

- 0 if all steps were successful.
- 1 if any error occurred

chowkidar

SYNOPSYS

badmailfrom usage
chowkidar [-V] [-f filename] [-b badmailfrom_file] [-B] -n count
chowkidar [-r] [-V] [-v] -q [-b badmailfrom_file]

badrcptto usage
chowkidar [-V] [-f filename] [-t badrcptto] [-T] -n count
chowkidar [-r] [-V] [-v] -q [-t badrcptto_file]

spamdb usage
chowkidar [-V] [-s spamdb] [-S] -n count
chowkidar [-r] [-V] [-v] -q [-s spamdb]

DESCRIPTION

chowkidar is a utility to help detect potential spammers by counting senders or recipients in qmail-send's deliver log file. The name is derived from the language Hindi, which means 'guard'. It looks for either of the following two entries in the qmail-send's log file.

@400000003d95f1d71571b154 info msg 181462: bytes 403 from <root@indimail.org> qp 8087 uid 0
@400000003d95f1d71571b155 starting delivery 145: msg 523306 to local manny@technology.indimail.org

chowkidar has two modes of operation

1. build mode – The first form of usage in synopsis shown above.
2. synchronize mode – The second form of usage in synopsis shown above.

In *build mode*, **chowkidar** maintains a hash table and keeps on incrementing the count against each sender found in the log file. After reading the log file completely, it prints out on stderr, the spammer's email id and count of mails received from the spammer. The default output file is the control file '***badmailfrom***'. This can be either changed by the BADMAILFROM environment variable or by the **'-b'** option. The **'-b'** option overrides the environment variable. If the output file is the qmail's ***badmailfrom*** control file, it appends the spammer's email id to the file. If the output file is anything other than '***badmailfrom***', the count of emails against the spammer is also appended to the file. Entries already present in the control files '***badmailfrom***' and '***spamignore***' are ignored and not maintained in the hash table. The entries in '***spamignore***' can have shell wild cards for pattern matching. If an entry starts with a '@' sign, all emails from the domain following the '@' sign will be ignored and not be treated as spammers. The hash table has a fixed compile time size of 5000, which can be changed by defining the ENVIRONMENT variable MAXADDR.

In *synchronize mode*, **chowkidar** synchronizes the contents of the badmailfrom file with a master copy maintained on ***hostcntrl***. It updates the master copy with entries which are more recent in the badmailfrom file. It also updates the badmailfrom file with entries which are more recent in the master copy. The master copy is the table '***badmailfrom***' if the filename is ***badmailfrom***; and the table '***spam***' if the filename is anything other than ***badmailfrom***. The default badmailfrom file is "***badmailfrom***" which can be changed by setting the **'-b'** option. This mode allows multiple hosts to synchronize their copies of ***badmailfrom*** with a central master copy and also replicate changes across all participating hosts running **chowkidar**.

The same logic applies to ***badrcptto*** control file. Except that **'-b'** gets replaced with **'-t'** and **'-B'** gets replaced with **'-T'**. In this case, count is done against the recipients. The default file '***badrcptto'*** can be changed by the '**-t**' option.

The same logic applies to ***spamdb*** control file. Except that **'-b'** gets replaced with **'-s'** and **'-B'** gets replaced with **'-S'**. In this case, count is done against the recipients. The default file '***spamdb'*** can be changed by setting the '**-s**' option.

Administrators can set up **chowkidar** in **cron(1)** to undertake automated anti-spam measures. Depending on the qmail-send's log file generation rate, the frequency can be every 30 min for a light system, every 2 hours for a medium system and every 4-6 hours for a heavy system. For a heavy system, remember to set the MAXADDR environment variable appropriately before running **chowkidar**. i.e. If the incoming rate is around 1 million mails/day and you are running **chowkidar** every 6 hrs, set MAXADDR to 250000 (mail_count_per_day * frequency / 24). The tool is especially useful if you are running multiple hosts and you want a spammer detected on any one of the hosts to be reflected on all hosts (synchronize mode of operation).

OPTIONS

-v

Sets verbose option

-V

Display Version Number

-f filename

The qmail deliver log file. If not given, the file defaults to stdin.

-b output_file

file in *badmailfrom* format. If not given, the environment variable BADMAILFROM is used. If the environment variable is also not defined, the badmailfrom file defaults to the qmail control file '*badmailfrom*'

-B

If **'-b'** option is not used this can be set to indicate badmailfrom format.

-t output_file

file in *badrcptto* format. If not given, the environment variable BADRCPTTO is used. If the environment variable is also not defined, the badrcptto file defaults to the qmail control file '***badrcptto***'

-T

If **'-t'** option is not used, this can be set to indicate badrcptto format. i.e. recipients are to be blocked (spammers generally get lot of bounces). A socially responsible ISP for example can use this option to prevent their own users from spamming.

-s

Input should be SMTPD's log file. The output file will be ***spamdb***. Additionally, **qmail-spamdb** will be run to create ***spamdb.cdb***.

-n count

Mail count from senders - above which the sender should be treated as a spammer.

-r

Synchronize mode of operation. This mode synchronizes local badmailfrom/badrcptto/spamdb file with a master copy maintained on ***hostcntrl***.

-q

quite mode. This mode will not display the list of spammers on stderr.

RETURN VALUE

Returns 0 for success, 1 for any failure.

logmonitor

SYNOPSYS

logmonitor [options]

DESCRIPTION

logmonitor is a utility for monitoring specific values in any log file being generated continuously by applications. It feeds the items to another utility called **hashtable** and **displaytop**.

hashtable creates a hash table of all lines fed to it and displays count of occurrences of each item feed to it.

displaytop sorts the entries and displays the entries on the screen with the entries having the maximum occurrences at the top.

OPTIONS

--logfile
Path of log file to monitor

--display_size
Maximum number of entries to display on screen.

--format
Type of format of the log file. i.e qmail-send or qmail-smtpd log file

--threshold
Minimum no of occurrences of an item above which the item should be reported

--follow
Continuously monitor the log file (like tail -f)

--batchmode
Set this if the output is not a terminal.

RETURN VALUE

0 for success, 1 for any failure.

resetquota

SYNOPSYS

resetquota [options] Maildir

DESCRIPTION

resetquota sets/corrects maildir quota for a Maildir. Without the -q option, resetquota calculates the disk usage for ***Maildir.*** It is usually used in two situations.

1. For correcting the disk usage used by a Maildir.
2. For setting maildir quota specification for a Maildir

OPTIONS

-v
Sets verbose operation

-u username
Use username for setting the ownership of the file.

-g group
Use group for setting group permission on the file.

-q quota_spec
Sets Maildir specification for the Maildir (See maildirquota(1)).

-p perm
Sets permission given by perm on the file.

Maildir
The full path of a Maildir.

RETURN VALUE

0 for success, 1 for any failure.

qmail-todo

SYNOPSYS

qmail-todo

DESCRIPTION

qmail-todo addresses a problem known as the silly qmail (queue) problem. This problem is found only on systems with high injection rates. qmail with a big local and remote concurrency could deliver a tremendous amount of messages but normally this can not be achieved because **qmail-send** becomes a bottleneck on those high volumes servers. **qmail-send** preprocess all new messages before distributing them for local or remote delivering. In one run qmail-send does one todo run but has the ability to close multiple jobs. Because of this layout **qmail-send** can not feed all the new available (local/remote) delivery slots and therefor it is not possible to achieve the maximum throughput. This would be a minor problem if one **qmail-send** run could be done in extreme short time but because of many file system calls (fsync and (un)link) a todo run is expensive and throttles the throughput.

qmail-todo tries to solve the problem by moving the todo routine into an external program. This reduces the run time in **qmail-send**. **qmail-todo** prepares incoming messages for local and remote delivering by creating

info/<messid>
local/<messid>
remote/<messid>

and removing
todo/<messid>).

As next **qmail-todo** transmits the <messid> to **qmail-send** which will add this message into the priority queue which schedules the message for delivery. **qmail-todo** will be started by qmail-start and therefor no additional setup is needed.

Communication between **qmail-send** and **qmail-todo**

qmail-todo -> qmail-send

D[LRB]<mesgid>\0
: Start delivery for new message with id <messid>.
the character L, R or B defines the type
of delivery, local, remote or both respectively.

L<string>\0
: Dump string to the logger without adding additional \n or similar.

qmail-send -> qmail-todo

H Got a SIGHUP reread ~/control/locals and ~/control/virtualdomains

X Quit ASAP.

qmail-todo sends "\0" terminated messages whereas **qmail-send** just send one character to qmail-todo.

IndiMail and Virtual domains

IndiMail has an idea of email domains that are "local" and "virtual". Local domains are ones which primarily match against /etc/passwd. Virtual domains match against domains listed in the qmail control file "virtualdomains". IndiMail makes use of the qmail users/assign file and virtualdomains file. The users/assign file gets compiled into a users/cdb file. It is a hashed database to speed searches for patterns. If a pattern is matched then IndiMail delivers the email to the directory defined in the file using the uid and gid which as also defined. IndiMail makes use of this method to have qmail deliver all virtual domain email as the single uid/gid indimail/indimail. It also uses it to direct delivery to a indimail/domains/<virtualdomain> directory.

Once **qmail-local** gets the information from the users/assign file it performs standard .qmail file processing in the directory. Normal .qmail-<user> files can be used for forwarding, aliases or invoking programs such as ezmlm. If no matches are found qmail-local looks for a .qmail-default file. This is the last stage in qmail-local's delivery mechanism. **qmail-local** uses this file to invoke the **vdelivermail** program. This program takes two parameters, the first is not used (it is there for backward compatibility). The second parameter is the default delivery if a virtual domain user can not be found. Basically, it can be a directory to deliver the email to, an email address to forward the email to or the string "bounce-no-mailbox" to bounce the mail back to the sender.

Once vdelivermail is started up, it uses the core IndiMail api calls to check for a virtual domain user. If the user exists, the email is delivered into their directory. If IndiMail was configured for hard quotas (default is yes with 5 Meg quota), then the size of the users current email files in their Maildir/new and Maildir/cur directories and other folders (excluding Trash) are counted. If the user is over quota the email is bounced back to the user with a bounce message that can be customized. If the message is 1Kbytes or smaller the email will always be delivered. This is so system administration programs can always get a message through to the user.

Converting current user accounts

The vconvert program can convert email accounts from one format into another format. Conversion can be between /etc/passwd, vpasswd files, MySQL small version) and MySQL (large version.

Most current IndiMail users would probably be interested in how to convert current domains into MySQL domains. To make it simple to convert an entire machine to

MySQL, use the following command: vconvert -c -s This will go through all the domains in ~indimail/domains directory and read each vpasswd file and load the contents into the indimail.indimail MySQL table. The vpasswd file is left untouched for safety. Vconvert can also be run against one or more domains at a time. This is done by running the command as so: vconvert \c \s domain1 domain2 ...

To convert all users (except root and system accounts) into a MySQL domain run the following command: vconvert -e -s domain. This reads all /etc/passwd accounts and creates MySQL entries using their passwords. The passwords can be in either /etc/passwd or /etc/shadow. These passwords should work under vchkpass authentication program.

Directory structure

Overall IndiMail directory structure

IndiMail gets it's own home directory. Under this directory there are the following:

bin - contains all the binaries
sbin – contains binaries required by administrators.
lib - contains the libindimail.a file
include - contains the C header files
users - for backward compatibility for people who mix /etc/passwd users with IndiMail users in one domain
domains - where all the virtual domains are kept.

Virtual domain user directory structure

IndiMail uses an adaptive directory structure based on a state file "***.dir-control***" which is automatically managed by the core IndiMail api functions "**vadduser**" and "**vdeluser**". The basic idea is to break up the user Maildir directories across multiple directories and sub directories so that there are never more than 100 user directories in a single directory.

In a single file system, the default directory setup allows for 62 directories in 3 levels and 100 user directories per directory. The total number of user directories is equal to (62 * 100) + (62 * 62 * 100) + (62 * 62 * 62 * 100) = over 24 million directories per filibuster. This should be more than sufficient for any site and probably goes beyond the technology of directory structures.

If you are going to be storing large numbers of user directories, make sure you set your file system to have a higher than normal percentage of inodes.

IndiMail will automatically create these directories and sub directories as needed and populate each directory with up to 100 user accounts. As soon as a directory reaches 100 users it will create the next directory or sub directory and store the new users

directory there.

You can use the **vreorg** program to recreate the ***.dir_control.***

B. Related Packages

B.1. ucspi-tcp

qmail's SMTP server doesn't run as a stand alone daemon. A helper program such as inetd, xinetd, or tcpserver runs as a daemon. When it receives a TCP connection to port 25, the SMTP port, it executes a copy of qmail-smtpd.

Inetd is the standard network server "super-server". It can be configured through /etc/inetd.conf to run qmail-smtpd, but the recommended tool is tcpserver, which is part of the ucspi-tcp package. ucspi-tcp is an acronym for UNIX Client-Server Program Interface for TCP, and it's pronounced ooks-pie tee see pee.
tcpserver is preferred over inetd because:

- tcpserver allows one to limit the number of simultaneous connections to a service. Inetd has a connection-rate limiting mechanism that temporarily disables services that are "too" busy.
- tcpserver can be configured to deny access to certain hosts or to recognize local hosts and flag them so qmail-smtpd can treat them differently.
- tcpserver is the only server supported by the author of qmail.

The source is available from ftp://cr.yp.to/ucspi-tcp/ucspi-tcp-0.88.tar.gz.

Gerrit Pape distributes the documentation for ucspi-tcp as man pages from http://smarden.org/pape/djb

ucspi-tcp was written by Dan Bernstein, who maintains a web page for it at http://cr.yp.to/ucspi-tcp.html.

B.2. rblsmtpd

If you've never been spammed, consider yourself very lucky. Most e-mail users are all to familiar with Unsolicited Bulk E-mail (UBE), aka "spam". Most of it is advertisements for sex sites, chain letters, and other scams. Back in the days of old, up until around 1998 or so, most MTA's on the Internet were open relays, i.e., they would accept mail from anyone for anyone, even if neither sender nor recipient was local. Spammers use open relays, if they can find any, to deliver their spam. It covers their tracks, redirects the backlash toward the "innocent" relay site, and saves them lots of CPU time and network bandwidth.

Such open relays are considered very bad form these days, and several anti-spam vigilante groups have created a mechanism for identifying open relays and other common sources of spam so they can avoid accepting SMTP connections from them.

rblsmtpd is an RBL SMTP Daemon. It sits between tcpserver and qmail-smtpd and rejects connections from systems identified on one of these lists.

For example, to run rblsmtpd under tcpserver, try something like:

```
#!/bin/sh
QMAILDUID=`id -u qmaild`
NOFILESGID=`id -g qmaild`
MAXSMTPD=`cat /var/indimail/control/concurrencyincoming`
exec /usr/local/bin/softlimit -m 2000000 \
/usr/local/bin/tcpserver -v -R -H -l 0 -x /etc/tcp.smtp.cdb -c "$MAXSMTPD" \
   -u $QMAILDUID -g $NOFILESGID 0 smtp /var/indimail/bin/rblsmtpd\
   -r relays.ordb.org /var/indimail/bin/qmail-smtpd 2>&1
```

rblsmtpd was previously available as a separate utility, but is now bundled with ucspi-tcp.

rblsmtpd was written by Dan Bernstein, who maintains a web page for it at http://cr.yp.to/ucspi-tcp/rblsmtpd.html.

B.3. fetchmail

IndiMail was designed for systems with full time, high speed connectivity. fetchmail is a set of tools that make IndiMail better suited to intermittent, low speed connectivity.

For outgoing mails, with fetchmail on such a system, IndiMail is configured to deliver all remote mail to a single maildir. The **maildirsmtp** command is used to upload the maildir to the ISP's mail hub when the connection is brought up. If the ISP supports QMTP (see QMTP under Advanced Topics), **maildirqmtp** can also be used. Alternatively, **qmail-remote** (with authenticated SMTP enabled) can be used to push the mails to the ISP's mail hub. **qmail-remote** configured with authenticated SMTP can also be used in situations where you have a dialup connection which allocates dynamic IP addresses.

For incoming mails, IndiMail can be used on the ISP side of the connection to implement AutoTURN or ETRN: an SMTP connection by a client causes the server to initiate a connection back to the client for sending messages queued on the server for the client. For clients who do not have static IP addresses, ODMR (On Demand Mail Relay) can be used. For ODMR, the client can use fetchmail to fetch mail. fetchmail can alternatively download mails from the ISP using any of the commonly used mail retrieval protocols in use on the internet (IMAP, POP3, ETRN, AUTOTURN, ODMR, etc).

fetchmail uses .fetchmailrc in the user's home directory. Here's a sample .fetchmailrc for a user on a qmail system:

```
poll mail.example.net proto pop3 nodns
   user dsill with password flubgart is dave here
```

fetchall forcecr

This instructs fetchmail to connect to mail.example.net via POP3, log in as user dsill, password flubgart, retrieve all messages, and deliver them to dave@localhost. The forcecr causes fetchmail to end each line with a carriage return when injecting the message on the local system via SMTP. qmail requires this.

IndiMail uses the serialmail package available from http://cr.yp.to/software/serialmail-0.75.tar.gz.

serialmail was written by Dan Bernstein, who maintains a web page for it at http://cr.yp.to/serialmail.html.

B.4. Pictures

There is a series of files in /var/indimail/doc with names starting with PIC. These are textual "pictures" of various situations that qmail handles. They show the flow of control through the various modules, and are very helpful for debugging and creating complex configurations.

Filename	Scenario
PIC.local2alias	locally-injected message delivered to a local alias
PIC.local2ext	locally-injected message delivered to an extension address
PIC.local2local	locally-injected message delivered to a local user
PIC.local2rem	locally-injected message delivered to a remote address
PIC.local2virt	locally-injected message delivered to an address on a local virtual domain
PIC.nullclient	a message injected on a null client
PIC.relaybad	a failed attempt to use the local host as a relay
PIC.relaygood	a successful attempt to use the local host as a relay
PIC.rem2local	a message received via SMTP for a local user

These files are also available on-line from http://www.qmail.org/man/index.html

If you want real pictures of qmail, check out Andre Opperman's "big qmail picture" at http://www.nrg4u.com

B.5. Error Messages

qmail error messages and what they mean.

See RFC 1893 for an explanation of the error codes in parentheses.

This appendix is incomplete.

qmail-local

- "qmail-local: usage: qmail-local [-nN] user homedir local dash ext domain sender aliasempty qqeh"
- "Out of memory. (#4.3.0)"
- "Unable to rewind message. (#4.3.0)"
- "Aack, child crashed. (#4.3.0)"
- "Unable to fork: reason. (#4.3.0)"
- "Unable to read message: reason. (#4.3.0)"
- "File has been locked for 30 seconds straight. (#4.3.0)"
- "Unable to open filename: reason. (#4.3.0)"
- "Recipient's mailbox is full, message returned to sender. (#5.2.2)"
- "Unable to chdir to maildir. (#4.2.1)"
- "Timeout on maildir delivery. (#4.3.0)"
- "Unable to read message. (#4.3.0)"
- "Temporary error on maildir delivery. (#4.3.0)"
- "Unable to open filename: reason. (#4.2.1)"
- "Unable to read message: reason. (#4.3.0)", 0
- "Unable to write filename: reason. (#4.3.0)", 0
- "Unable to run /bin/sh: reason. (#4.3.0)"
- "Unable to forward message: reason."
- "This message is looping: it already has my Delivered-To line. (#5.4.6)"
- "Unable to stat home directory: reason. (#4.3.0)"
- "Uh-oh: home directory is writable. (#4.7.0)"
- "Home directory is sticky: user is editing his .qmail file. (#4.2.1)"
- "Warning: home directory is sticky.", 0
- "Uh-oh: .qmail file is writable. (#4.7.0)"
- "Unable to switch to homedir: reason. (#4.3.0)"
- "Sorry, no mailbox here by that name. (#5.1.1)"
- "Uh-oh: first line of .qmail file is blank. (#4.2.1)"
- "Uh-oh: .qmail has file delivery but has x bit set. (#4.7.0)"
- "Uh-oh: .qmail has prog delivery but has x bit set. (#4.7.0)"
-

qmail-smtpd

- "252 Cannot VRFY user, but will accept message and attempt delivery (#2.7.0)"
- "421 Service not available, closing tranmission channel (#4.3.2)"
- "450 atrn service unavailable (#5.7.1)"
- "451 Requested action aborted: out of memory (#4.3.0)"
- "451 Requested action aborted: unable to read controls (#4.3.0)"
- "451 Requested action aborted: unable to figure out my IP addresses (#4.3.0)"
- "451 Requested action aborted: database error (#4.3.2)"
- "451 Requested action aborted: qqt failure (#4.3.0)"
- "451 Requested action aborted: timeout (#4.4.2)"

- "451 Requested action aborted: Bare LF received. (#4.6.0)"
- "451 Requested action aborted: DNS temporary failure (#4.4.3)"
- "451 Requested action aborted: problem with child and I can't auth (#4.3.0)"
- "451 Requested action aborted: child won't start and I can't auth (#4.3.0)"
- "451 Requested action aborted: unable to open pipe and I can't auth (#4.3.0)"
- "451 Requested action aborted: unable to write pipe and I can't auth (#4.3.0)"
- "451 Unable to queue messages (#4.3.0)"
- "451 Unable to queue messages, status <exit_status> (#4.3.0)"
- "453 No message waiting for node(s) <domain>"
- "500 command not recognized (#5.5.1)"
- "501 invalid parameter syntax (#5.3.2)"
- "501 auth exchange cancelled (#5.0.0)"
- "501 malformed auth input (#5.5.4)"
- "502 unimplemented (#5.5.1)"
- "503 Polite people say hello first (#5.5.4)"
- "503 you're already authenticated (#5.5.0)"
- "503 no during mail transaction (#5.5.0)"
- "503 auth not available (#5.3.3)"
- "503 MAIL first (#5.5.1)"
- "503 RCPT first (#5.5.1)"
- "503 bad sequence of commands (#5.3.2)"
- "504 auth type unimplemented (#5.5.1)"
- "530 authentication required (#5.7.1)"
- "535 authorization failed (#5.7.0)"
- "550 sorry, bounce messages should have a single envelope recipient (#5.7.1)"
- "550 sorry, sender account <recipient> is absent (#5.1.1)"
- "550 sorry, sender account <recipient> is inactive (#5.2.1)"
- "550 sorry, sender account <recipient> is overquota (#5.2.2)"
- "552 sorry, that message size exceeds my databytes limit (#5.3.4)"
- "553 Invalid mail address, must have a domain part (#5.1.8)"
- "553 Bad sender's system address (#5.1.8)"
- "553 sorry, that domain isn't allowed to be relayed thru this MTA with masquerading (#5.7.1)"
- "553 sorry, that domain isn't allowed to be relayed thru this MTA without authentication (#5.7.1)"
- "553 sorry, your envelope recipient is in my badrcptto list (#5.7.1)"
- "553 we don't relay (#5.7.1)"
- "553 sorry, your envelope sender is in my badmailfrom list (#5.7.1)"
- "553 <domain> etrn service unavailable (#5.7.1)"
- "553 atrn service unavailable (#5.7.1)"
- "554 too many hops, this message is looping (#5.4.6)"
- "555 syntax error in address (#5.1.3)"

qmail.c

- "communication with mail server failed (#4.4.2)"

- "connection to mail server rejected (#4.4.1)"
- "connection to mail server timed out (#4.4.1)"
- "envelope address too long for qq (#5.1.3)"
- "mail server permanently rejected message (#5.3.0)"
- "mail server temporarily rejected message (#4.3.0)"
- "Message contains banned attachment (#5.7.1)"
- "Message contains virus (#5.7.1)"
- "qq crashed (#4.3.0)"
- "qq internal bug (#4.3.0)"
- "qq out of memory (#4.3.0)"
- "qq permanent problem (#5.3.0)"
- "qq read error (#4.3.0)"
- "qq temporary problem (#4.3.0)"
- "qq timeout (#4.3.0)"
- "qq trouble creating files in queue (#4.3.0)"
- "qq trouble creating pipes (#4.3.0)"
- "qq trouble creating temporary files (#4.3.0)"
- "qq trouble getting uids/gids (#4.3.0)"
- "qq trouble in home directory (#4.3.0)"
- "qq trouble making network connection (#4.3.0)"
- "qq unable to read configuration (#4.3.0)"
- "qq waitpid surprise (#4.3.0)"
- "qq write error or disk full (#4.3.0)"
- "SPAM or junk mail threshold exceeded (#5.7.1)"
- "temporary problem with SPAM filter (#4.3.0)"
- "unable to exec filter (#4.3.0)"
- "unable to exec qq (#4.3.0)"
- "unable to fork filter (#4.3.0)"
-

spawn.c

- "Internal error: delnum negative. (#4.3.5)"
- "Internal error: delnum too big. (#4.3.5)"
- "Internal error: delnum in use. (#4.3.5)"
- "Internal error: messid has nonnumerics. (#5.3.5)"
- "Internal error: messid too long. (#5.3.5)"
- "Internal error: messid too short. (#5.3.5)"

B.6. Credits

The content in this book borrows verbatim and heavily from material available at www.qmail.org, www.cr.yp.to, The Big qmail Picture by André Oppermann and Dave Sill's excellent book Life With Qmail

B.7. Licensing

This document is covered by the OpenContent License, version 1.0. See http://www.opencontent.org/opl.shtml for the full license. Basically, you can copy, redistribute, or modify this document provided that modified versions, if redistributed, are also covered by the OpenContent License.

OpenContent License (OPL)

Version 1.0, July 14, 1998.

This document outlines the principles underlying the OpenContent (OC) movement and may be redistributed provided it remains unaltered. For legal purposes, this document is the license under which OpenContent is made available for use.

The original version of this document may be found at http://opencontent.org/opl.shtml

LICENSE

Terms and Conditions for Copying, Distributing, and Modifying

Items other than copying, distributing, and modifying the Content with which this license was distributed (such as using, etc.) are outside the scope of this license.

1. You may copy and distribute exact replicas of the OpenContent (OC) as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the OC a copy of this License along with the OC. You may at your option charge a fee for the media and/or handling involved in creating a unique copy of the OC for use offline, you may at your option offer instructional support for the OC in exchange for a fee, or you may at your option offer warranty in exchange for a fee. You may not charge a fee for the OC itself. You may not charge a fee for the sole service of providing access to and/or use of the OC via a network (e.g. the Internet), whether it be via the world wide web, FTP, or any other method.

2. You may modify your copy or copies of the OpenContent or any portion of it, thus forming works based on the Content, and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified content to carry prominent notices stating that you changed it, the exact nature and content of the changes, and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the OC or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License, unless otherwise permitted under applicable Fair Use law.

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the OC, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a

whole which is a work based on the OC, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it. Exceptions are made to this requirement to release modified works free of charge under this license only in compliance with Fair Use law where applicable.

3. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to copy, distribute or modify the OC. These actions are prohibited by law if you do not accept this License. Therefore, by distributing or translating the OC, or by deriving works herefrom, you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or translating the OC.

NO WARRANTY

4. BECAUSE THE OPENCONTENT (OC) IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE OC, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE OC "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK OF USE OF THE OC IS WITH YOU. SHOULD THE OC PROVE FAULTY, INACCURATE, OR OTHERWISE UNACCEPTABLE YOU ASSUME THE COST OF ALL NECESSARY REPAIR OR CORRECTION.

5. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MIRROR AND/OR REDISTRIBUTE THE OC AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE OC, EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES