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.**
- **pam-multi –** pam authentication module which provides generic authentication making it possible for any external program to authenticate against IndiMail's database.
- **nssd** – provides Name Service Switch extending user/group lookups beyond the systems existing databases like passwd, group

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/web

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

You can also get titbits about IndiMail at

twitter	http://twitter.com/indimail
facebook	http://www.facebook.com/pages/indimail/88063516703?_fb_noscript=1
source	https://sourceforge.net/apps/laconica/indimail/

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 the files section at sourceforge and familiarize yourself with installation instructions. For further details, you may want to read section 2.3.2 below.

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
badrcptto	none	qmail-smtpd	blacklisted "To" addresses
badrcptpatterns	none	qmail-smtpd	Same as above except that wildcard patterns can be given
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

Control	*Default*	*Used By*	*Purpose*
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
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.

Control	*Default*	*Used By*	*Purpose*
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
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

Control	*Default*	*Used By*	*Purpose*
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
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	

Control	*Default*	*Used By*	*Purpose*
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	
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
203.	INCLUDE_TRASH	qmail-send	used	If set, vdelivermail will include trash in quota calculations.

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	/home/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
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

Delivery	*Attempts*	*Seconds D-HH:MM:SS*
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
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:

<table>
<tr><td>qmail-smtpd</td><td rowspan="2">--></td><td rowspan="2">qmail-queue</td><td rowspan="2">--></td><td rowspan="2">qmail-send
qmail-clean</td><td rowspan="2">--></td><td>qmail-rspawn</td><td rowspan="2">--></td><td>qmail-remote</td></tr>
<tr><td>qmail-inject</td><td>qmail-lspawn</td><td>qmail-local</td></tr>
</table>

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

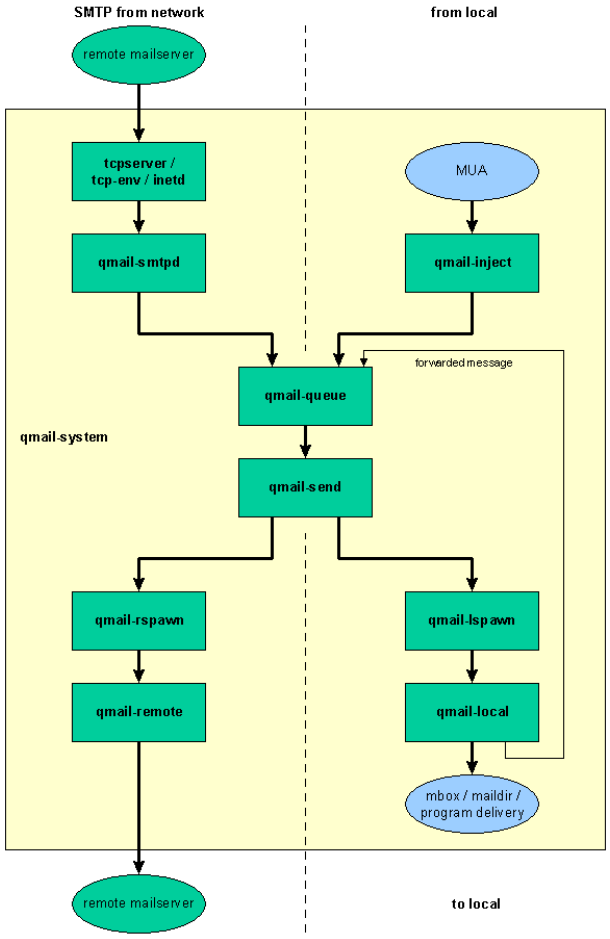
Currently info/457 serves two purposes: first, it records the envelope sender; second, its modification time is used to decide when a message has been in the queue too long. In the future info/457 may store more information. Any non-backwards-compatible changes will be identified by version numbers.

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

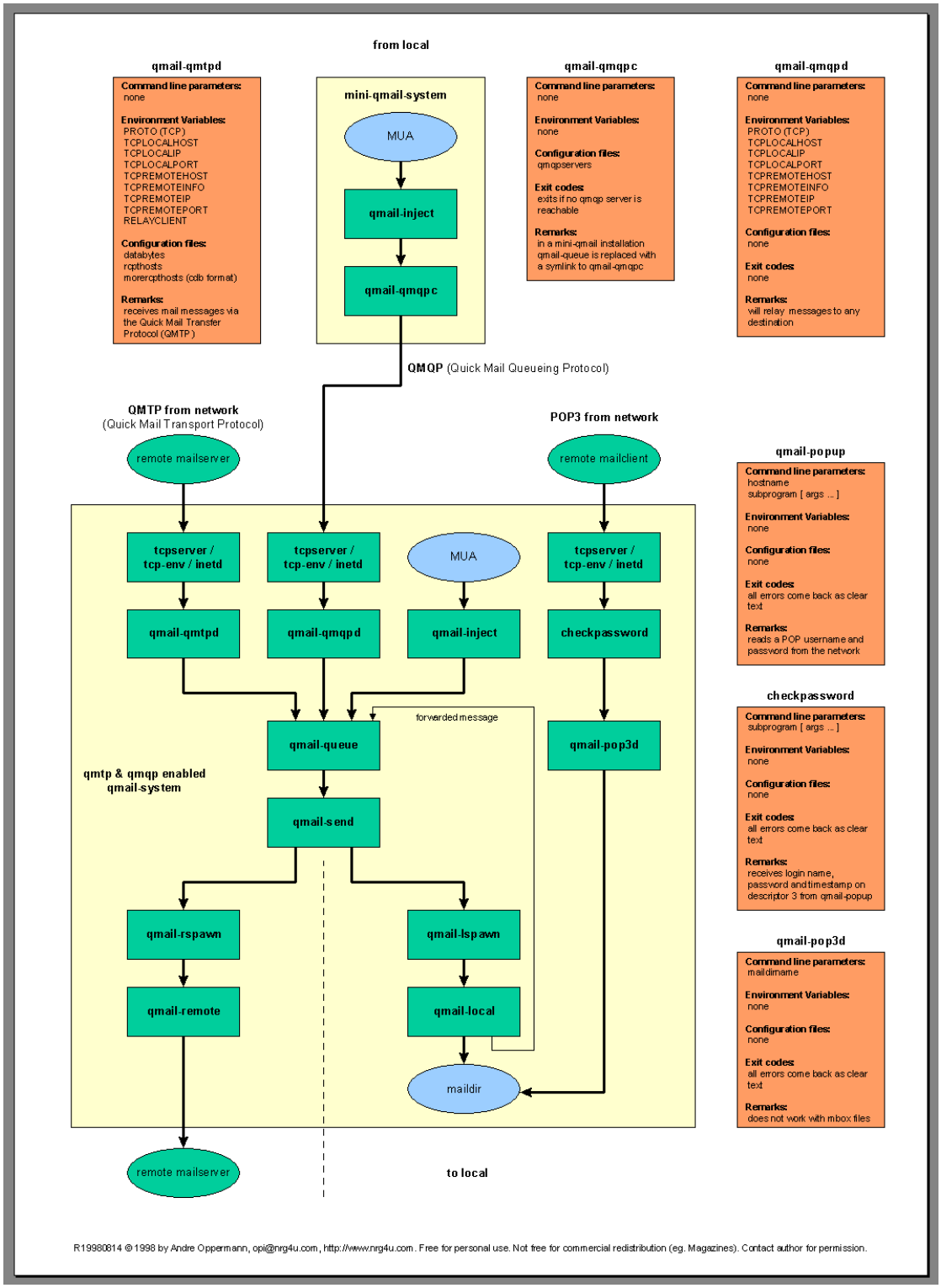
qmail-local

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



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 new message 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 ...

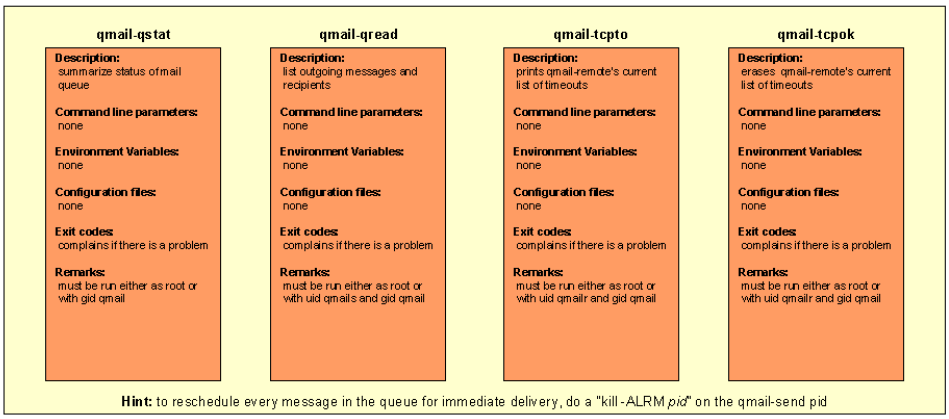
Environment Variables:
none

Configuration files:
none

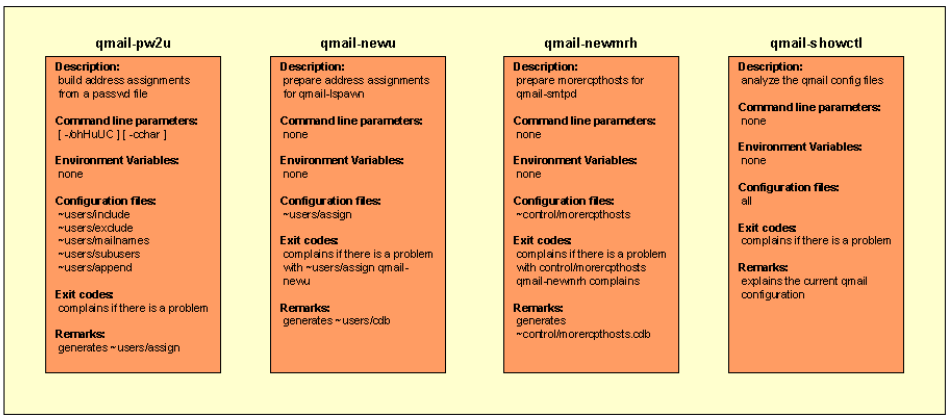
Exit codes:
none

Remarks:
reads the body of the message from its standard input

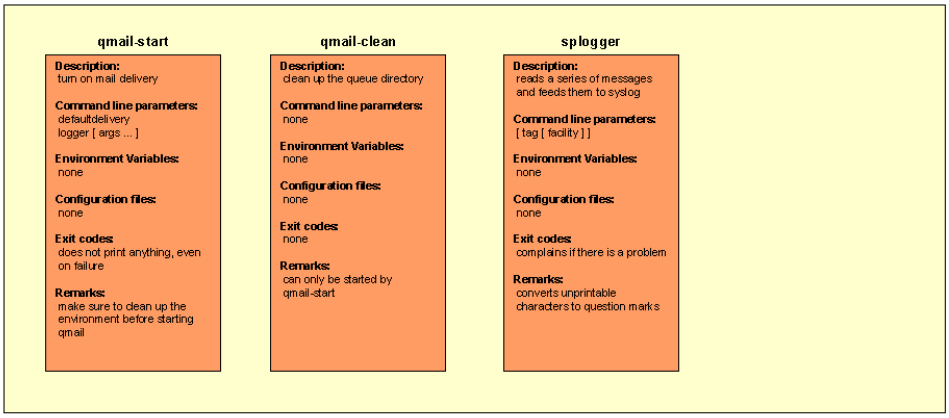
queue management



user and system management



other qmail daemons



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.

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