Qpopper

Administrator's Guide

Qpopper Version 4.0

This document is designed for use online. The Contents and Index sections plus all cross-references and URLs are "hot linked." In the Contents section, click the topic, and the topic information appears. In the Index, click the page number. Also, clicking any cross-reference displays the topic referenced. To open any URL (such as a web page or email address), just click it.

Acknowledgments

Qpopper is the work of many individuals and organization around the world. The original work was done at UC Berkeley around 1991 and many have contributed since. QUALCOMM has maintained, enhanced, and distributed Qpopper since 1993. While we don't have an accurate collection of the names of all, their contributions are appreciated. Qpopper 4.0 was developed by Randall Gellens. This guide was written by Randall Gellens, Gigi Miller, Scott Anthony, and Armand Rouleau. Thanks to all the users Qpopper, whose suggestions have made it a much better program than it otherwise would have been.

November 2001
PM80-48468-1x1

Contents

Introducing Qpopper

Introduction

Welcome to Qpopper! Qpopper enables a UNIX/Linux machine to act as a Post Office Protocol version 3 (POP) server. POP allows email clients, such as Eudora, to pick up the user's messages. Qpopper is the most widely-used POP server.

Important. To use Qpopper, you should have a comprehensive knowledge of UNIX/Linux systems and how they operate.

Qpopper is normally used with standard UNIX mail transfer and delivery agents such as Sendmail or Smail. Qpopper follows UNIX/Linux conventions for mailbox (spool) location, locking, etc.

This server is fully compliant with RFC 1939 (which defines the POP protocol) and RFC 2449 (which defines the POP extension mechanism and the extended response codes), and works with all known POP clients, such as Eudora.

Qpopper supports the latest standards and provides a large number of features, such as bulletins, enhanced support for wireless devices, Authenticated Post Office Protocol (APOP), integration with Pluggable Authentication Modules (PAM) and packages such as Kerberos versions 4 or 5, Dynamic Relay Access Control (DRAC), etc. Qpopper also works with security packages such as OpenSSL or Certicom's SSL Plus to provide Transport Layer Security/Secure Sockets Layer (TLS/SSL) encryption of all traffic to and from the email client. Qpopper 4.0 includes enhanced performance features and easier administration.

You can always get the latest release of Qpopper at <http://www.qpopper.org>.

This manual applies to Qpopper 4.0.

File Names and Commands in this Manual

In this manual, actual file and directory names appear `like this`. If the file name includes a variable part (such as a user name), the variable part is in italics, for example, a temporary spool file is called `.`*`user`*`.pop`. Commands you type are formatted **`like this`**.

Foundation

Qpopper's goals are security, stability, safety, features, and performance.

Qpopper has multiple levels of protection against common security vulnerabilities, such as buffer overruns.

Qpopper takes extra precautions to guard against spool corruption, even if there is a system crash or power failure during an update of the software.

You can also indicate when some precautions are not required, for example, if your users do not access mail using shell accounts. This allows Qpopper to run at maximum performance levels.

Getting Help

Technical Support

Many of your questions can be answered by looking in this guide, particularly in the Troubleshooting section. In addition, the Qpopper web site at <http://www.qpopper.org> has a variety of resources, including a Frequently Asked Questions (FAQ) page.

Mailing Lists

An unofficial public mailing list exists for Qpopper administrators, such as yourself. To subscribe, type the word **subscribe** as the body of an email message and send it to: **qpopper-request@lists.pensive.org**. This mailing list also receives official announcements from QUALCOMM about Qpopper.

There is also a low volume mailing list that exists solely for official announcements from QUALCOMM about Qpopper. To subscribe to it, type the word **subscribe** as the body of an email message and send to: **qpopper-announce-request@rohan.qualcomm.com**

Other Resources

There is a man page for Qpopper. To display it, type **man qpopper** on the command line. However, all the information in the man page, and much more, is contained in this document.

To make suggestions to improve Qpopper, write to qpopper-suggest@qualcomm.com

To report bugs in Qpopper, write to qpopper-bugs@qualcomm.com

Installing and Setting Up Qpopper

What you need to do

Before you can install Qpopper on your UNIX/Linux operating system, you must install and configure an SMTP server and local delivery agent, such as Sendmail or Smail.

Installing Qpopper

Important. In the mail spool directory, some systems have symbolic links from `/usr/mail` to `/usr/spool/mail`. Make sure you check this before installing Qpopper.

To install Qpopper, do the following:

1. Download the Qpopper software from `<http://www.qpopper.org>`. This is a single file (called a *distribution*) which contains the source code, scripts, and documentation in a compressed format (also known as a compressed archive or *tarball*). The name of the file is `qpopper`, followed by the release, an optional `-no-test`, and ends in `.tar.gz`. The `.tar` means this is a single file that contains multiple files and directories, while the `.gz` means this single file also is compressed to save space. If you choose the name with `-no-test`, it is a smaller version that omits the *test* directories and scripts.

   The distribution for Qpopper 4.0 is called `qpopper4.0.tar.gz`. or `qpopper4.0-no-test.tar.gz`

2. Uncompress the files by typing **`gunzip qpopper4.0.tar.gz`**

3. Unpack the files by typing **`tar xvf qpopper4.0.tar`**. The unpacked files will then be placed in a new directory called `qpopper4.0`.

4. Change to the `qpopper4.0` directory by typing **`cd qpopper4.0`**.

5. On the command line, type **`./configure`**.

   Add any other configure flags as needed or desired.

6. On the command line, type **`make`**. Running *make* builds Qpopper for your operating system.

7. Copy the resulting Qpopper executable to a public location. The executable is called `popper` and can be found in the `popper` subdirectory. Normally, typing **`make install`** does this for you.

   Although there is no required location, many system administrators prefer `/usr/local/lib`, and this is used in example lines (such as `/etc/inetd.conf` lines).

If you modify Qpopper, be sure to edit the file `popper/banner.h` and add a string.

Make Commands

- **`make`**

  Compiles Qpopper. If using APOP (see “APOP” on page 49), also compiles `popauth`. If *--enable-poppassd* used with `./configure`, also compiles `poppassd` in the `password` directory.

  Note that you must run **`./configure`** before **`make`**.

- **`make install`**

  Copies the Qpopper executable and man pages to a standard location. If using APOP, also copies `popauth`. If *--enable-poppassd* specified with `./configure`, also copies `poppassd`.

  You need to be user `root` to do this.

- **`make clean`**

  Deletes all executables and compiled object code.

- **`make realclean`**

  Deletes all executables and compiled object code, plus configuration information and temporary files. Use this before **`./configure`** when you want to be certain that you are starting off fresh.

Uninstalling Qpopper

To uninstall Qpopper, remove the `inetd.conf` line, and delete the Qpopper executable and directories.

Configuring Qpopper

You can run Qpopper in either Internet daemon (*inetd*) mode or standalone mode. In *inetd* mode, Qpopper is controlled by inetd. Standalone mode requires that you leave the server running all the time.

If using standalone mode, configure your system to launch Qpopper at start-up. See section “Enabling Standalone Mode” on page 43 for more information.

If you are not going to use standalone mode, modify your `/etc/inetd.conf` file to contain the line below. You may have to modify it to include any desired command-line flags and where you decided to place the Qpopper executable, if not `/usr/local/lib`:

```
pop3 stream tcp nowait root /usr/local/lib/popper qpopper -s
```

Note. If you have a great number of users connecting the server, you may need to increase the `inetd` timeout value to prevent it from assuming that popper is looping, which otherwise causes it to kill Qpopper and write a log entry noting *service looping*. You can increase the global *inetd* timeout by passing *inetd* a command line argument. On some systems, you can alter the timeout for Qpopper by changing `nowait` to `nowait.`*`timeout`*, for example, `nowait.400`.

If your OS does not have an `inetd.conf` file, then it may use the configuration file `/etc/servers`. Enter the following option:

```
pop3 tcp /usr/local/lib/popper qpopper -s
```

On all systems, your `/etc/services` file needs to include the following line:

```
pop3         110/tcp                # Post Office
```

Note. Be sure to remove (or comment out) any lines from `/etc/services` which have the same port (110) and remove (or comment out) any corresponding lines from `/etc/inetd.conf` or you may have problems, such as *address already in use* errors.

Restart *inetd* with a **`kill -HUP inetdpid`** (some systems can use **`inetd -c`**).

On Linux, you can use **`ps x | grep inetd`** to find the PID of *inetd*. On many flavors of UNIX, you can use **`ps -e | grep inetd`**. On AIX, use **`refresh -s inetd`** Note that you need to be user `root` to do this.

If you are running NIS, please don't forget to update your maps.

Note that you also need to copy the man pages. Generally, this requires that you copy `man/*.8` to `/usr/man/man8`. Normally, typing **`make install`** does this automatically.

Configure Options

The following options (flags) can be passed to `./configure` to enable compile-time changes:

An italicized variable after an = sign indicates that a specific value may be entered here. In most cases, if you do not enter a specific value, a default value is used. For example, if you see **---enable-*apop=path,*** *path* indicates that you may specify the path, such as: **--enable-*apop= /etc/pop.auth***

Many compile-time options can be overridden at run-time by setting a configuration file option, or in some cases, a command-line flag.

Type this option...	This happens...	Configuration File Option
--disable-status **--enable-status** (default)	When disabled, Qpopper does not update the read/unread status of messages (a feature relied on by some mail clients) and also does not save the message's unique identifier (UID). This forces the UID for every message to be recalculated, using more CPU but potentially less I/O. See "Performance" on page 57 for more information.	update-status-headers
--disable-old-spool-loc **--enable-old-spool-loc** (default)	When disabled, Qpopper doesn't check for old *`.user`*`.pop` files in old locations when HASH_SPOOL or HOMEDIRMAIL is used. This speeds things up.	check-old-spool-loc

Type this option...	This happens...	Configuration File Option
--enable-any-kerberos-principal **--disable-any-kerberos-principal** (default)	Enabling permits use of any Kerberos principle in client authentication.	
--enable-apop=*path* **--disable-apop** (default)	Enable this option to use APOP. This value is the location of the authorization database. Users found in this database may not use user/pass authentication (since they are found in the APOP database) unless you use the *-p* command-line flag or the *clear-text-password* configuration file option to authorize it. The default path is `/etc/pop.auth`. See “APOP” on page 49 for instructions and configure options.	
--enable-auth-file=*path* **--disable-auth-file** (default)	Enable this option to specify a file that permits only users listed in the file to have Qpopper access. The format is a list of user names, one per line.	auth-file
--enable-auto-delete **--disable-auto-delete** (default)	When enabled, Qpopper automatically and unconditionally deletes messages that have been downloaded using the RETR command (the normal command for accessing messages). **Caution.** This option could result in lost mail. Be sure to inform your users that the option is in effect before enabling it.	auto-delete

Type this option...	This happens...	Configuration File Option
--enable-bulldb=*path* **--disable-bulldb** **(**default)	When enabled, Qpopper uses a central database located in the bulletin directory (instead of a file in the user's home directory) to keep track of which bulletins the user has seen. This feature uses the user's `.popbull` file the first time for backwards compatibility. Specify the path if an alternate location for the bulletin database is desired. See “Using Bulletins” on page 53 for details. Also see “Performance” on page 57 for more information. Enabling this option also enables bulletins in general.	
--enable-bulletins=*path* **--disable-bulletins** (default)	Enable bulletins and sets the path for the bulletin directory (default is */var/spool/bulls*). See “Using Bulletins” on page 53 for details. This is the compiled value as opposed to the command line or configuration file option that enables bulletins. This makes bulletins the default regardless of the command line or configuration file options.	bulldir
--enable-debugging **--disable-debugging** (default)	Enables verbose logging when used with the *-d* or *-t* command line flag or the *debug* or *tracefile* configuration file option. Enable this if you are having problems figuring out why Qpopper is not working. Remember that it can generate a lot of output, and log files on busy systems expand quickly.	
--enable-cache-dir=*path* **--disable-cache-dir** (default)	Specifies the location of the user’s cache files, which are used to speed up Qpopper performance in server mode. The default is the same directory as for temporary spool files (see *--enable-temp-drop-dir*).	cache-dir

Type this option...	This happens...	Configuration File Option
--disable-check-hash-dir **--enable-check-hash-dir** (default)	When disabled, Qpopper doesn't check for or create the hashed spool directories. Disable if you precreate the directories. Has no effect either way unless you also use *--enable-hash-spool.*	check-hash-dir
--disable-check-pw-max **--enable-check-pw-max** (default)	Disabling prevents Qpopper from checking for expired passwords.	check-password-expired
--enable-group-bulls **--disable-group-bulls** (default)	Shows bulletins by groups (group name is second element in bulletin name). See “Using Bulletins” on page 53.	group-bulletins

--enable-hash-spool=*1*
--enable-hash-spool=*2*
--disable-hash-spool (default)
	When enabled, the subdirectory for the mail spools is determined from the user name by either hashing the first four characters or by using directories equal to the first letter and the second letter (if any). For example, if the spool directory is `/var/mail`, the spool file for user `maida` would be:
	`/var/mail/maida`
	HASH_SPOOL not set
	`/var/mail/o/maida`
	HASH_SPOOL=1
	`/var/mail/m/a/maida`
	HASH_SPOOL=2
	See “Performance” on page 57 for more information.
		hash-spool

--enable-home-dir-mail=*file* **--disable-home-dir-mail** (default)	If mail is spooled into the user's home directory, set this to be the correct file name for your system. The default file name is `.mail`.	home-dir-mail
--enable-home-dir-misc **--disable-home-dir-misc** (default)	Causes the `.user.pop` and the `.user.cache` files to be placed in the user's home directory.	home-dir-misc

Type this option...	This happens...	Configuration File Option
--enable-keep-temp-drop **--disable-keep** (default)	Keep the `.user.pop` file (the temporary drop file). Normally this file is deleted when the session ends. Some sites like to retain it to determine the last time a user has accessed his or her mail.	keep-temp-drop
--enable-ksockinst **--disable-ksockinst** (default)	Uses `getsockinst()` for Kerberos instance.	
--enable-kuserok **--disable-kuserok** (default)	Uses `kuserok()` to vet Kerberos users.	
--enable-log-facility=*name* **--disable-log-facility**	Specifies the log facility. Default is LOG_LOCAL0 or LOG_MAIL, depending on the operating system. Note that the value you pick must exist in the system's `log.h` file, and be typed exactly as it appears there.	log-facility
--enable-log-login **--disable-log-login** (default)	When defined, Qpopper logs successful authentications.	log-login
--enable-low-debug **--disable-low-debug** (default)	Sets the _DEBUG macro (not the same as simply DEBUG) for low-level debugging. Also turns off compiler optimizations. Don't do this unless you know what you're doing.	
--enable-new-bulls=*number* **--disable-new-bulls** (default)	New users receive only the single newest bulletin by default. This value sets the number of bulletins for new Qpopper users. For example, pass **`--enable-new-bulls=10`** to give new users a maximum of ten bulletins. See section "Using Bulletins" on page 53 for details.	max-bulletins
--enable-nonauth-file=*path* **--disable-nonauth-file** (default)	Define this value to specify a file that excludes listed users. The format is a list of user IDs, one per line. Users appearing in the file are unable to use Qpopper.	nonauth-file

Type this option...	This happens...	Configuration File Option
--enable-old-uidl **--disable-old-uidl** (default)	Generates message unique identifiers (UIDS) using old (pre-3.x) style encoding. This is useful only if you also use *--disable-status*, have existing users with old (pre-3.x) spool files, and you want to keep the UIDS the same.	old-style-uid
--disable-optimizations **--enable-optimizations** (default)	Turns off all compiler optimizations. This is generally only useful when running under a debugger.	
--enable-popbulldir=*path* **--disable-popbulldir** (default)	Specifies an alternate location for users' .`popbull` files. See section “Using Bulletins” on page 53 for details.	
--enable-poppassd **--disable-poppassd** (default)	Running *make* creates the `poppassd` executable as well as *popper*. This is a password-changing daemon	
--enable-popuid=*pop* **--disable-popuid** (default)	This value is the username of the owner of the APOP database. See “APOP” on page 49 for details.	
--enable-scram=*path* **--disable-scram** (default)	Includes scram capability in the APOP database file (default path is */etc/pop.auth*). This is not recommended.	
--enable-secure-nis-plus **--disable-secure-nis-plus** (default)	Add this definition if you are running Secure Network Information Systems (NIS+).	
--enable-server-mode **--disable-server-mode** (default)	Enables server mode by default. See section “Enabling Server Mode” on page 41 for details. Also see “Performance” on page 57 for more information.	server-mode

Type this option...	This happens...	Configuration File Option
--enable-server-mode-group-exclude=*group* **--disable-server-mode-group-exclude** (default)	Sets server mode to *off* for users who are members of the specified group. See section “Enabling Server Mode” on page 41 for details. Also “Performance” on page 57 for more information.	group-no-server-mode
--enable-server-mode-group-include=*group* **--disable-server-mode-group-include** (default)	Sets server mode to *on* for users who are members of the specified group. It may also be helpful to disable shell access to these users, thus ensuring that only Qpopper and the local delivery agent access the user’s spool file. See section “Enabling Server Mode” on page 41 for details. Also see “Performance” on page 57 for more information.	group-server-mode
--enable-shy **--disable-shy** (default)	Enable if you don't want Qpopper to display its version in the POP protocol banner or CAPA IMPLEMENTATION response of unauthenticated users. Some sites believe this improves security since it avoids advertising that an old version (perhaps with known vulnerabilities) is being used. Others feel is makes the site more likely to be attacked, since it also avoids advertising when running a secure version.	shy
--enable-specialauth **--disable-specialauth**	This needs to be enabled if your system supports special authorization mechanisms like shadow passwords or special crypt programs. However, this is usually set correctly by `./configure`. Disable this if you keep passwords in `/etc/passwd`.	

Type this option...	This happens...	Configuration File Option
--enable-spool-dir=*directory*	Allows you to specify the directory for mail spool files, such as `/var/mail` or `/var/spool/mail`. If not specified, `./configure` searches for it.	spool-dir
--enable-standalone **--disable-standalone** (default)	Creates a standalone daemon instead of one to be run out of *inetd*. You can specify an IP address and/or port number to bind to as **`parameter 1`**, for example, **`popper 199.46.50.7:8110 -S or popper 8110 -S -T600`**. If not specified, the IP address defaults to all that are available. The default port is 110, except when _DEBUG (not simply DEBUG) is defined, when it is 8765. See “Enabling Standalone Mode” on page 43 for more information. Also see “Performance” on page 57 for more information.	
--enable-temp-drop-dir=*path* **--disable-temp-drop-dir**	Specifies an alternate directory for temporary mail drop files. The default is the spool directory. See “Performance” on page 57 for more information.	temp-dir
--enable-timing **--disable-timing** (default)	Writes timing information to log at session end. Includes whole seconds (elapsed) for authentication, initialization, and cleanup.	timing
--enable-uw-kludge **--disable-uw-kludge** (default)	Checks for and hides status messages created by University of Washington software.	uw-kluge

Type this option...	This happens...	Configuration File Option
--with-drac=*lib-path* **--without-drac** (default)	Enables use of DRAC. Specify path to DRAC libraries or leave blank if installed in usual place. DRAC is a method of authorizing SMTP sessions for IP addresses which have recently authenticated using POP. This can be useful for clients which do not support SMTP AUTH, but the long term solution is SMTP AUTH.	
--disable-update-abort **--enable-update-abort** (default)	By default, Qpopper enters *update* state on a session abort. Disabling this option causes Qpopper to ignore any deletions if the session is aborted. Note that RFC 1939, section 6 prohibits the Qpopper default behavior, but experience showed that otherwise users on noisy lines were often unable to delete their mail. Disable this option to inhibit the default behavior, and obey RFC 1939.	update-on-abort

Type this option...	This happens...	Configuration File Option
--with-kerberos5=*directory* **--without-kerberos5=*directory*** (default)	Enable to use Kerberos version 5 libraries. Or, use this and also define KRB4 if you want to use Kerberos version 4 libraries. For Kerberos 4, you must also update the `makefile` so you can load the appropriate libraries (*-lkrb*). This option works only if you have Kerberos headers and libraries installed. If you want to use the `kuserok()` function to vet users, pass *--enable-kuserok* to `./configure`. If you want to accept any Kerberos principal in the client request, pass *--enable-any-kerberos-principal*. If your system has the `getsockinst()` function, pass *--enable-ksockinst*. It retrieves the IP address of the local end of a connection and looks up the host name. Kerberos then uses that host name. This allows Kerberos to use different instances for different virtual addresses on the same machine. You can also define KERBEROS_SERVICE to specify which Kerberos service to use. The default is `rcmd`. However, the use of `pop` is common. The *-K service* command-line or *kerberos-service* configuration file option can be used instead of the KERBEROS_SERVICE define. You can remove KRB5_KRB4_COMPAT from `config.h` if you are using the Kerberos 5 libraries and want to disable backwards compatibility with a Kerberos 4 server. Define NO_CROSSREALM if desired. You can obtain a Kerberos engine for your system from the MIT Kerberos Distribution Page. Go to the following site: `<http://web.mit.edu/kerberos/www/index.html>`	

Type this option...	This happens...	Configuration File Option
--with-openssl=path **--without-openssl** (default)	Uses the OpenSSL library for SSL/TLS support. You can obtain the OpenSSL library and utilities from <http://www.openssl.org>	
--with-sslplus=path **--without-sslplus** (default)	Uses the SSL Plus library from Certicom for SSL/TLS support.	
--with-gdbm **--without-gdbm**	Normally, `./configure` uses *gdbm* if installed on the system. You can override this by using this option.	
--with-sslplus-crypto =path **--without-sslplus-crypto** (default)	Uses Security Builder® from Certicom for TLS/SSL cryptography.	
--with-pam= *servicename* **--without-pam *(***default)	Define if you want to use PAM for authentication. The default service name is `pop3`.	
--enable-warnings **--disable-warnings** (default)	Enables additional compiler warnings. Use this if you don't trust the Qpopper authors :-).	

Run-Time Command Line Options

You can set Qpopper run-time options either from the command line or a configuration file.

Some systems have limitations on the length or number of command line options in `inetd.conf`. (For example, the Solaris INETD.CONF man page states that no more than five options are permitted.) You can use configuration files for run-time options to get around this limit or to set options on a per-user basis. Also, some run-time options can only be set from a configuration file. Use the *-f*, *-u,* and *-U* command-line options to cause Qpopper to use a configuration file. See the following section "Run-Time Options from a Configuration File" on page 27 for details.

In the following table are the run-time options and their descriptions for Qpopper:

Values in *italics* indicate a variable, for example, in **-D *drac-host***, *drac-host* is the variable.

Enter the name or IP address of the DRAC host after *-D*. The option might be:
`-D foo.example.org`

Type this option...	Description	Equivalent Configuration File Option
-b *bulldir*	Specific location of the bulletin directory. Overrides the compiled value, if any. See “Using Bulletins” on page 53 for more information.	bulldir
-B	If compiled with *--enable-bulldb* (see **--enable-bulldb**) in the “Configure Options” on page 11, this option allows sessions to proceed even if the bulletin database can't be opened. This allows users to get their mail, but may mean some users won't see bulletins for some time or even at all.	bulldb-nonfatal
-c	Changes uppercase user names to lowercase. This permits users to configure their clients with user names in UPPER or MiXeD case. They can still login assuming their actual user name is all lowercase.	downcase-user
-C	When set, domains are trimmed from user names before use. For example, if a user named *maida* enters her login name in her POP client as *maida@example.org*, Qpopper treats this as just *maida*.	trim-domain
-d	Enables debug logging if *--enable-debugging* used with `./configure`). Output is in *syslog*. If this option is used, it should be first, so that debug records are generated for subsequent options.	debug
-D *drac-host*	If `./configure` used with *--with-drac*, this option specifies the DRAC host. Defaults to `localhost`.	drac-host

Type this option...	Description	Equivalent Configuration File Option
-e login_delay=*delay*, expire=*expire*	Sets POP3 extensions. This option announces the *login-delay* and expire response tags to the CAPA command. By default the expire field is *expire never*, and the login delay is 0. Except for *expire 0* or *expire never*, these are not enforced by Qpopper. Sysadmins have to implement them by some other means.	announce-expire announce-login-delay
-f *config-file*	Reads additional run-time options from the specified file. See the following section “Run-Time Options from a Configuration File” on page 27 for option names and syntax. **Caution.** There are no restrictions on which options may appear in files specified with the *-f* command-line flag or the *config-file* configuration file option in files chained from *-f*, so be certain that the specified file is not writable by users.	config-file
-F	When updating the spool at the end of a session, this option instructs Qpopper to rename the temporary file to the spool instead of copying it. This reduces I/O at session end by a third, but is likely to break programs such as `biff` or the shell's mail check feature. **Use this option only if such programs are not used.** It is safest to only enable this option when users do not have shell access to the mail server. See “Performance” on page 57 for more information.	fast-update
-k	If *--enable-kerberos5* used with `./configure`, this option enables Kerberos support.	kerberos

Type this option...	Description	Equivalent Configuration File Option
-K *service name*	If *--enable-kerberos5* used with `./configure`, this option specifies the Kerberos service to use (same as the compile time *kerberos_service* define). The default is `rcmd`, although the use of `pop` is popular.	kerberos-service
-l 0 **-l 1** **-l 2** Only available when `./configure` was run with *--with-openssl* or *--with-sslplus*.	Specifies TLS/SSL support. •**0** The default. TLS/SSL is not supported. •**1** Enables support for the STLS command. This permits TLS/SSL negotiations on the standard (or any) port, allowing the same port to be used by TLS/SSL and regular clients. •**2** Enables alternate-port TLS/SSL. Some older clients require this. (The usual port for alternate-port TLS/SSL with pop is 995.)	tls-support
-L *lock-refresh*	Checks if the mail lock needs to be refreshed every this many messages. You normally do not need to adjust this. See “Performance” on page 57 for more information.	mail-lock-check

Type this option...	Description	Equivalent Configuration File Option
-p 0 **-p 1** **-p 2** **-p 3** **-p 4**	Sets clear text handling options when APOP is available. •**0** The default. Clear text passwords are permitted for all users except those in the APOP database (who must use APOP). •**1** Clear text passwords are never permitted for any user (users not in the APOP database cannot login). •**2** Clear text passwords are always permitted (even if an APOP entry exists), which allows them to be used as a fallback. •**3** They are permitted on the local interface (127.*.*.*) only. •**4** Permits clear text passwords only if TLS/SSL has been negotiated for the session. See “Security and Authentication” on page 49 for more information.	clear-text-password
-R	Disables the reverse lookups on client IP addresses.	reverse-lookup
-s	Enables statistics logging. After each session ends, a statistics record is written to the log. This record resembles the following example: `randy 0 0 1 385 randy.examp.org 192.168.2.4` in which Username: *randy* Deleted messages: *0* Deleted octets: *0* Messages left on server: *1* Octets left on server: *385* Name of client machine: *randy.examp.org* IP address of client machine: *192.168.2.4*	statistics
-S	Enables server mode by default. See “Enabling Server Mode” on page 41 for details.	server-mode

Type this option...	Description	Equivalent Configuration File Option
-t *logfile*	Enables debug logging if *--enable-debugging* used with `./configure`. Output is written to the file specified as *logfile*. If this option is used, it should be first, so that debug records are generated for subsequent options. You can also use this when you didn't pass *--enable-debugging* to `./configure`, which causes log messages to be written to *logfile* instead of *syslog*.	tracefile
-T *timeout*	You can change the timeout for client reads. The default is 120 seconds. Qpopper terminates the connection with the client if no input is received in this many seconds. RFC 1939 states that this timeout should be 600 seconds (10 minutes). However, ideal settings in some cases are between 30 and 120 seconds.	timeout
-u	Reads additional run-time options from a file named `.qpopper-options` in the user's home directory, if present. See the following section "Run-Time Options from a Configuration File" on page 27 for option names and syntax.	user-options
-U	Same as above, but the file is called *`user`*`.qpopper-options` and is in the spool directory. **This file should not be owned by nor writable by the user.** Use this to set options on a user-specific basis that you don't want the user to change.	spool-options
-y	Specifies the log facility that Qpopper uses. Note that this does not apply to popauth, nor to the daemon in standalone mode. Both of these continue to use the compile-time default.	log-facility

Run-Time Options from a Configuration File

You can set Qpopper run-time options either from the command line or in a configuration file.

Configuration files use different option names and a different syntax than the command-line (because command-line options are limited to one character).

The general syntax of the config file (in Backus-Naur form (BNF)) is:

```
config-line         ::= comment-line / reset-line / set-line
comment-line        ::= "#" <comment-text to end of line>
reset-line          ::= "reset" boolean-option
set-line            ::= implicit-set / explicit-set
explicit-set        ::= "set" option "=" value
implicit-set        ::= "set" boolean-option
option              :: = boolean-option / integer-option / string-option / mnemonic option
value               ::= "true" / "false" / integer / string / mnemonic
string              ::= “1*255CHAR”
CHAR                ::= any printable character except space or tab
```

In other words, a line starts with *set* or *reset*, then an option name, and either ends there (for Boolean options) or has an equal sign (=) followed by a value. You can put spaces or tabs around each element.

A comment line starts with a pound sign (#). The rest of the line is ignored. (Everything else on the line is a comment.)

Note that *reset* can be used only with boolean options. The = and the value are omitted when *reset* is used. When *set* is used with a boolean option, you can omit the = and value if you wish (it defaults to *true*), or you can use any of the four values *true*, *false*, *1*, or *0*.

Some options have restrictions indicating that they can't be used in a `.qpopper-options` file in a user's home directory and/or in a `.`*`user`*`.qpopper-options` file in the spool directory.

In the following table are the command line options you can use:

Note that an italicized variable after a = sign indicates that the correct value can be entered here.

Type this option...	Description	Equivalent Command Line Option
announce-expire	This integer option announces the EXPIRE response tags to the CAPA command. This option is permitted in all configuration files. However, except for *expire 0* or *expire never*, it is not enforced by Qpopper. Sysadmins have to implement them by some other means. By default, the expire field is *expire never*.	-eexpire=*xx*
announce-login-delay	This integer option announces the LOGIN-DELAY response tag to the CAPA command. A minimum login delay is not currently enforced by Qpopper. This option is permitted in all configuration files. Default: none	-elogin_delay=*xx*
auto-delete	This Boolean option, when set, causes Qpopper to automatically and unconditionally deletes messages that have been downloaded using the RETR command (the normal command for accessing messages). **Caution.** This option could result in lost mail. Be sure to inform your users that the option is in effect before enabling it.	none

Type this option...	Description	Equivalent Command Line Option
bulldb-max-retires Only valid if `./configure` run with *--enable-bulldb*	Sets the maximum number of attempts to lock the bulletins database. You normally do not need to adjust this. The value should only be changed if you know if your system has `usleep(3C)` or not. On systems with `usleep(3C)`, this can be a large value (the default is 75). On systems without `usleep(3C)`, this should remain small (the default is 10). This integer option is permitted in all configuration files. Default: > 5 or 10	none
bulldb-nonfatal Only valid if compiled with *--enable-bulldb*	If `./configure` was run with *--enable-bulldb* (see **--enable-bulldb** in the “Configure Options” on page 11), this option allows sessions to proceed even if the bulletin database can't be opened. This allows users to get their mail, but may mean some users won't see bulletins for a time or even at all. This Boolean option is permitted in all configuration files. Default: false	-B
bulldir	Specific location of the bulletin directory. Overrides the compiled value, if any. See “Using Bulletins” on page 53 for more information. This is a string option and is permitted in all configuration files.	-b *bulldir*
cache-dir	Set this to the full path to the directory for cache files if you do not want user cache files to be in the same directory as temporary drop files. Note the use of `/tmp` is not recommended because a system reboot deletes these files. Default: Temp drop directory	none

Type this option...	Description	Equivalent Command Line Option
cache-name	The name of the cache file. You should not normally set this option. Default: `.%s.cache`	none
chunky-writes	By default, Qpopper aggregates data to be sent to clients into large chunks. This may be faster or slower, depending on the specifics of both the client and server hardware and networking stacks as well as network elements in between (such as routers). Also, some networking stacks do their own aggregation. Under congested network conditions, larger packets increase the effects of lost packets and thus client or server timeouts, which may cause *POP timeout* or *EOF* errors. When TLS/SSL is in effect, smaller packets increase the overhead needed to send data, which may result in worse performance. You can adjust the Qpopper behavior by setting this option. The values are: • **default** Always send large chunks • **always** Same as *default* • **never** Never aggregate data into large chunks • **tls** Only aggregate data into large chunks when TLS/SSL has been negotiated for the session •**ss** Same as *tls* Default: default	none

Type this option...	Description	Equivalent Command Line Option
clear-text-password	Sets clear text handling options. The permitted values are: • **default** Clear text passwords are permitted for all users, except those in the APOP database (who are thus required to use APOP). • **never** Clear text passwords are never permitted. Users not in the APOP database cannot use Qpopper. • **always** Clear text passwords are always permitted, even for users in the APOP database. This allows them to be used as a fallback when an APOP client is temporarily not available. • **local** Clear text passwords are permitted on the local interface only (127.*.*.*). • **tls** Clear text passwords are permitted when TLS/SSL has been negotiated for the session. • **ssl** (same as *tls*). The *tls* and *ssl* values are only valid if *--with-openssl* or *--with-sslplus* was used with `./configure`. This mnemonic option is not permitted in a configuration file in the user's home directory nor in the spool directory. Default: default	-p 0 -p 1 -p 2 -p 3 -p 4

Type this option...	Description	Equivalent Command Line Option
config-file	Reads additional run-time options from the specified file. This string option is permitted in all configuration files. **Caution.** There are no restrictions on which options may appear in files specified with the *-f* command-line flag or the *config-file* configuration file option in files chained from *-f*. Be certain that the file specified with *-f* or in any files it chains to are not writable by users.	-f *config-file*
debug Only valid if `./configure` was run with *--enable-debugging*	Enables debug logging. Output is in *syslog*. If this option is used, it should be first, so that debug records are generated for subsequent options. This Boolean option is permitted in all configuration files. Default: false	-d *debug*
downcase-user	Changes uppercase user names to lowercase. This permits users to configure their clients with user names in UPPER or MiXeD case. They can then login, assuming their actual user name is all lowercase. This Boolean option is not permitted in configuration files specific to a user (those in a user's home directory or in the spool directory). Default: false	-c
drac-host Only valid if compiled with *--with-drac*	This option specifies the drac host. This string option is permitted in all configuration files. Defaults to `localhost`.	-D *drac-host*

Type this option...	Description	Equivalent Command Line Option
fast-update .	When updating the spool at the end of a session, this option instructs Qpopper to rename the temporary file to the spool instead of copying it. This reduces I/O at session end by a third, but is likely to break programs such as `biff` or the shell's mail check feature. **Use this option only if such programs are not used.** It is safest to only enable this option when users do not have shell access to the mail server. Default: false See “Performance” on page 57 for more information. This Boolean option is permitted in all configuration files.	-F
home-dir-mail	If mail is spooled into the user's home directory, set this to be the correct file name for your system. The default file name is `.mail`. This string option is not permitted in configuration files specific to a user (those in a user’s home directory or in the spool directory).	none
home-dir-misc	Causes the `.user.pop` and the `.user.cache` files to be placed in the user's home directory. This Boolean option is not permitted in configuration files specific to a user (those in a user’s home directory or in the spool directory).	none
kerberos Only valid if `./configure` run with *--enable-kerberos5*	This option enables Kerberos support. This Boolean option is not permitted in configuration files specific to a user (those in a user’s home directory or in the spool directory).	-k

Type this option...	Description	Equivalent Command Line Option
kerberos-service Only valid if `/configure` run with *--enable-kerberos5*	This option specifies the Kerberos service to use (same as the compile time KERBEROS_SERVICE define). The default is `rcmd`, although the use of `pop` is popular. This Boolean option is not permitted in configuration files specific to a user (those in a user's home directory or in the spool directory).	-K *service-name*
log-facility	Specifies the log facility that Qpopper uses. Note that this does not apply to popauth, nor to the daemon in standalone mode. These continue to use the compile-time default. Values are: • **mail** Qpopper logs to LOG_MAIL facility. • **local0** Qpopper logs to LOG_LOCAL0 facility. • **local1** Qpopper logs to LOG_LOCAL1 facility. • **local2** Qpopper logs to LOG_LOCAL2 facility. • **local3** Qpopper logs to LOG_LOCAL3 facility. • **local4** Qpopper logs to LOG_LOCAL4 facility. • **local5** Qpopper logs to LOG_LOCAL5 facility. • **local6** Qpopper logs to LOG_LOCAL6 facility. • **local7** Qpopper logs to LOG_LOCAL7 facility. Default: determined at compile time, usually local1 or mail, depending on the operating system.	y

Type this option...	Description	Equivalent Command Line Option
log-login	When set, Qpopper logs successful authentications using the specified string. Within the string, an occurrence of *%0* is replaced with the Qpopper version, *%1* with the user name, *%2* with the user's host name, and *%3* with the user's IP address. Default: none, unless *--enable-log-login* used with `./configure`, in which case *(v%0) POP login by user \%1\ at (%2) %3* is used.	none
mail-command	Set this to the full path to sendmail or other such program used to submit new messages. Qpopper uses this to implement XTND XMIT. The default is determined at compile time. An example is `/usr/sbin/sendmail`	none
mail-lock-check	Checks if the mail lock needs to be refreshed every this many messages. You normally do not need to adjust this. See “Performance” on page 57 for more information. This integer option is permitted in all configuration files. Default: 5000	-L *msgs*
max-bulletins	Specifies the maximum number of old bulletins seen by new users. Default: 1	none
no-atomic-open	When set, Qpopper uses a method of opening lock files that may work over NFS. This has not been thoroughly tested. Default: false	none

Type this option...	Description	Equivalent Command Line Option
reverse-lookup Sense reversed from command-line switch.	Disables the reverse lookups on client IP addresses. This Boolean option is not permitted in configuration files specific to a user (those in a user's home directory or in the spool directory). Default: true	-R **(Sense reversed)** Using -R is the same as *set reverse-lookup = false*.
server-mode	Enables server mode by default. See section server mode for details. This Boolean option is permitted in all configuration files. The default is determined at compile time.	-S
spool-dir	Set this to the full path to the mail spool directory. The default is determined at compile time. An example is `/var/spool/mail`	none
spool-options	Reads additional run-time options from a file named *`username`*`.qpopper-options` in the spool directory. This is a Boolean option and is not permitted in configuration file in the user's home directory.	-U

Type this option...	Description	Equivalent Command Line Option
statistics	Enables statistics logging. After each session ends, a statistics record is written to the log. This record resembles the following example: `stats randy 0 0 1 385 randy.example.org 192.168.2.4` in which Username: *randy* Deleted messages: *0* Deleted octets: *0* Messages left on server: *1* Octets left on server: *385* Name of client machine: *randy.example.org* IP address of client machine: *192.168.2.4* This is a Boolean option that is permitted in all configuration files. Default: false	-s
temp-dir	Set this to the full path to the directory to be used for temp drop files if you do not want `.user.pop` (temporary drop files) to be located in the spool directory. Note that use of `/tmp` is not recommended because a system reboot deletes these files. This could cause lost mail. Default: spool directory	none
temp-name	The name of the temporary drop files. You should not normally set the option. Default: `.%s.pop`	none

Type this option...	Description	Equivalent Command Line Option
timeout	You can change the timeout for client reads. Qpopper terminates the connection with the client if no input is received in this many seconds. RFC 1939 states that this timeout should be 600 seconds (10 minutes). However, ideal settings in some cases are between 30 and 120 seconds. This integer option is permitted in all configuration files. The default is 120 seconds.	-T *timeout*
tls-cipher-list Only valid when *--with-openssl* used with `./configure`	Specifies the permitted cipher suites. See the OpenSSL documentation for syntax. You normally do not need to set this. This string option is not permitted in a configuration file in the user's home directory nor in the spool directory.	none
tls-identity-file Only valid if `./configure` used with *--with-sslplus*	Specifies a file which contains the server's TLS/SSL certificate and encrypted private key. This string option is not permitted in a configuration file in the user's home directory nor in the spool directory.	none
tls-passphrase Only valid if `./configure` used with *--with-sslplus*	Specifies the passphrase to decrypt the server's private key (in the *identify* file). This string option is not permitted in a configuration file in the user's home directory nor in the spool directory.	none

Type this option...	Description	Equivalent Command Line Option
tls-private-key-file Only valid if `./configure` used with *--with-openssl*	Specifies a file which contains the server's TLS/SSL private key. **Note.** This private key must not be encrypted. If the private key is contained in the same file as the certificate (as specified with *tls-server-cert-file*), you do not need to specify this option. This string option is not permitted in a configuration file in the user's home directory nor in the spool directory.	none
tls-server-cert-file Only valid if `./configure` used with *--with-openssl*	Specifies a file which contains the server's TLS/SSL certificate. This file may also contain the server's unencrypted private key. This string option is not permitted in a configuration file in the user's home directory nor in the spool directory.	none
tls-support Only valid when *--with-openssl* or *--with-sslplus* used with `./configure`	Specifies TLS/SSL support. The permitted values are: •**default** TLS/SSL is not supported. •**none** (SAME AS **DEFAULT**) •**stls** Enables support for the STLS command. This permits TLS/SSL negotiations on the standard (or any) port, allowing the same port to be used by TLS/SSL and regular clients. •**alternate-port** Enables alternate-port TLS/SSL. Some older clients require this. (The usual port for alternate-port TLS/SSL with pop is 995.) This mnemonic option is not permitted in a configuration file specific to a user (in the user's home directory or in the spool directory.) Default: default	-l 0 -l 1 -l 2

Type this option...	Description	Equivalent Command Line Option
tls-version Only valid when *--with-openssl* used with `./configure`	Restricts the version of TLS/SSL recognized in session negotiations. You normally do not need to set this. Supported values are: •**default** (same as *SSLv23)* •**SSLv2** Forces Qpopper only to understand SSLv2 client hello messages. •**SSLv3** Forces Qpopper only to understand SSLv3 client hello messages. This especially means that it does not understand SSLv2 client hello messages, which are widely used for compatibility reasons. •**TLSv1** Forces Qpopper only to understand TLSv1 client hello messages. This especially means that it does not understand SSLv2 client hello messages, which are widely used for compatibility reasons. It also does not understand SSLv3 client hello messages. •**SSLv23** Allows Qpopper to understand SSLv2, SSLv3, and TLSv1 client hello messages. This is the best choice when compatibility is a concern. This is the default value. •**all** (same as *SSLv23)* This mnemonic option is not permitted in a configuration file in the user's home directory nor in the spool directory. Default: default	none

Type this option...	Description	Equivalent Command Line Option
tracefile	Enables debug logging if --*enable-debugging* used with `./configure`). All debug and standard log records are written to the specified file. If this option is used, it should be first, so that debug records are generated for subsequent options. If used without --*enable-debugging*, redirects all log messages to the specified file. This string option is permitted in all configuration files.	-t *logfile*
trim-domain	When set, domains are trimmed from user names before use. For example, if a user named *maida* enters her login name in her POP client as *maida@example.org*, Qpopper treats this as just *maida*. This Boolean option is not valid in configuration files specific to a user (those in a home directory or the spool directory). Default: false	-C
user-options	Reads additional run-time options from a file named `.qpopper-options` in the user's home directory, if present. This Boolean option is not permitted in a configuration file in the user's home directory nor in the spool directory. Default: false	-u

Enabling Server Mode

Server mode reduces Qpopper's I/O usage for sessions where users download and delete all their mail, or when they leave all their mail on the server (in other words, sessions where they delete all messages or no messages). Server mode can only be used when Qpopper and the local delivery agent are the only processes which alter the mail spool. **Do not enable server mode for users who telnet to the mail server and access the mail spool directly, or run local programs which access the mail spool.**

Server mode is ideal for situations where a system acts only as a mail server; where users do not have shell accounts.

You can enable or disable server mode on a user-by-user basis, by users' group membership, for all users, or any combination.

In normal (non-server) mode, Qpopper starts a session by locking the mail spool, copying all messages to a temporary file, zeroing the mail spool, then unlocking it. At the end of the session all non-deleted messages (plus any new ones that arrived during the session) are copied back to the spool. By working from a temporary copy of the mail spool, Qpopper operates at the highest safety level, at the cost of extra I/O.

In server mode, Qpopper starts a session by scanning the mail spool instead of copying it. It works from the mail spool during the session. If at the end of the session all messages are deleted, or no changes have been made to the state of the spool or its messages, Qpopper avoids having to do any copying at all.

Because most POP clients download and delete all messages by default, enabling server mode can save a lot of server I/O. Even when users turn on *leave mail on server* in their POP client, most sessions do not result in any changes, again potentially saving a lot of I/O.

Qpopper further reduces I/O at the start of a session by avoiding the initial scan of the spool in many cases where server mode is used. This can result in a mail-check time of well under a second, versus a minute or more for very large spools.

You can decrease the situations in which Qpopper needs to copy and update the spool at the end of a session by using the *--disable-status* configure flag, or the *update-status-headers* configuration file option. However, this increases CPU usage and prevents Qpopper from keeping track of which messages have been downloaded, a feature relied upon by some clients. This also prevents Qpopper from storing the unique identifier (UID) of each message in an *X-UIDL:* header, forcing it to recalculate it when needed. It also prevents Qpopper from writing *Status:* headers.

You can enable server mode on a per-user basis. Just set the default for all users at compile or run time. You can indicate that certain users will use Qpopper in server mode or not, regardless of the defaults.

To specify server mode as a default for all users at compile time, do the following:

- Run the configure script, adding *--enable-servermode*.

To specify server mode as a default for all users at run time, do the following:

- Use the *-S* run-time option or the *server-mode* option in a configuration file.

To indicate that users of a specific group should use server mode, regardless of the default, do the following:

- Run the configure script, adding *--enable-server-mode-group-include=group* where *group* is the name of the group, or use the *group-sever-mode* option in a configuration file. Users who are members of this group will use server mode. (It may also be convenient to disable shell access to users in this group, to prevent them from accessing the spool other than through POP).

To indicate that users of a specific group should **not** use server mode, regardless of the default, do the following:

- Run the configure script, adding *--enable-server-mode-group-exclude=group* where *group* is the name of the group, or use the *group-no-server-mode* option in a configuration file. Users who are members of this group will **not** use server mode.

To set server mode on or off for a specific user, do the following:

- Place a file called `.qpopper-options` in the user's home directory, set the *server-mode* option in this file, and use the *-u* command-line option or include the *user-options* option in a configuration file. You can also use a file called `.user.qpopper-options` in the spool directory, with the *-U* command-line option or *spool-options* configuration file option.

The determination of server mode is in the following order:

- Compile-time default (*--enable-server-mode*)
- Run-time default (*-S* or *server-mode*)
- Compile-time inclusion group (*--enable-server-mode-group-include=group*) that the user is a member of
- Run-time inclusion group (*group-server-mode*)
- Compile-time exclusion group (*--enable-server-mode-group-exclude=group*) that the user is a member of
- Run-time exclusion group (*group-no-server-mode*)
- User-specific `.qpopper-options` file in the home directory or `.user.qpopper-options` in the spool directory.

Enabling Standalone Mode

Normally, Qpopper is launched from *inetd* (or a similar program). However, you can build a version of Qpopper that operates as a standalone daemon if you prefer. Operating Qpopper in standalone mode may result in better performance, but you may lose the ability for access control and load throttling. To enable standalone mode, use the *--enable-standalone* flag with `./configure`.

To enable Qpopper's standalone mode, do the following:

1. On the command line, type **`./configure --enable-standalone`** (plus any other desired options)
2. Configure Qpopper to be launched on system start-up. If desired, specify the IP address and/or port number to bind to at run-time as parameter 1, for example, **`popper 199.46.50.7:8110 -S`** or **`popper 8110 -S -T600`**. If not specified, the IP address defaults to all available. The default port is 110 except when _DEBUG (not simply DEBUG) is defined, then it is 8765.

Using Macros

There are a large number of options which can be modified at compile time. The ones which are generally useful can be set by using a flag with `./configure`, for example *--enable-debugging* for DEBUG, which is the recommended way. Many of these can also be adjusted using run-time options.

The obscure ones have to be set manually. To do so, edit `config.h` after running `./configure`. To set a macro, add or change a `#define` line. For example, to set macro `FOO`, add `#define FOO`.

The following table lists Qpopper macros, a description and the corresponding configure flags used to set the macro:

Macro	Description	Configure flag
MMDF_SEP_CHAR Default is '\001' (<CTL>-A)	A line that starts with this character is considered an MMDF separator. The rest of the line is copied as the separator regardless of its value.	none
BULLDIR	See “Configure Options” on page 11 for a description.	--enable-bulldb=*path*
NEWBULLCNT	See “Configure Options” on page 11 for a description.	--enable-new-bulls=*number*
OFF_T	If OFF_T is not typed for you, then set this definition to a type that *lseek* and *ftruncate* and expect it as an offset parameter. UID_T, GID_T, and PID_T are also available for portability.	none
BLOCK_UID	Qpopper prevents access by users with a UID at or below this value. This is because normally low UIDs are only used for `root` and daemon processes, which should never be checking mail. Attempts to do so may indicate a security attack. The default value is 10. This indicates that `root` and other users with UID values of 10 and under are not able to login via Qpopper.	none
POP_FACILITY	See “Configure Options” on page 11 for a description.	--enable-log-facility=*name*

Macro	Description	Configure flag
KERBEROS	See “Configure Options” on page 11 for a description.	--with-kerberos5=*directory*
AUTHFILE	See “Configure Options” on page 11 for a description.	--enable-auth-file=*path*
NONAUTHFILE	See “Configure Options” on page 11 for a description.	--enable-nonauth-file=*path*
SPEC_POP_AUTH	See “Configure Options” on page 11 for a description.	--enable-specialauth
RPOP	This feature allows the POP client to use the `hosts.equiv` and `.rhosts` files for system/user validation. This feature is not recommended since it can easily be defeated and thus create a serious security risk.	none
SECURENISPLUS	See “Configure Options” on page 11 for a description.	--enable-secure-nis-plus
DEBUG	See “Configure Options” on page 11 for a description.	--enable-debugging
SETPROCTITLE	This definition controls whether `setproctitle()` should be used to display user, host, and status in the process title. Your operating system must support this feature. Set automatically by `./configure`.	none
KEEP_TEMP_DROP	See “Configure Options” on page 11 for a description.	--enable-keep-temp-drop
BIND43	BIND 4.3 is a domain name service. Set automatically by `./configure`.	none
SYSLOG42	Qpopper defaults to BSD 4.3 syslog.	none
HOMEDIRMAIL	See “Configure Options” on page 11 for a description.	--enable-home-dir-mail=Mailbox
APOP= "/etc/pop.auth"	See “Configure Options” on page 11 for a description.	--enable-APOP=*path*

Macro	Description	Configure flag
POPUID=pop	See “Configure Options” on page 11 for a description.	--enable-popuid=*pop*
GNUPASS	This definition specifies the use of the modified GNU `getpass()` routine which allows longer than 8 character passwords to be entered when using *popauth*. There is also a definition within *popauth (NO_GETLINE)* if you compile this code, and it complains if it doesn’t have a `getline` routine.	none
CONTENT_LENGTH	If your local delivery agent inserts a *Content-Length:* header into the message, define this option. This is usually set automatically for most systems.	none
SERVER_MODE	See “Configure Options” on page 11 for a description.	==enable-server-mode
NO_STATUS	See “Configure Options” on page 11 for a description.	--disable-status
NOUPDATEONABORT	See “Configure Options” on page 11 for a description.	--disable-update-abort
HASH_SPOOL=*(1|2)*	See “Configure Options” on page 11 for a description.	--enable-hash-spool
BULLDB	See “Configure Options” on page 11 for a description.	--enable-bulldb=*path*
CHECK_SHELL	Enable this compile time feature to lock out users via the shell value. A user shell of /POPPER/ANY/SHELL allows the user access but blocks other programs that check shells.	none
GDBM	This value uses the GNU GDBM library instead of NDBM.	none
_DEBUG	This permits popper to be run from a shell, for hard-core and heavy debugging.	--enable-low-debug

Macro	Description	Configure flag
LOG_LOGIN	See “Configure Options” on page 11 for a description.	--enable-log-login
AUTO_DELETE	See “Configure Options” on page 11 for a description.	--enable-auto-delete
SHY	See “Configure Options” on page 11 for a description.	--enable-shy
USE_PAM	See “Configure Options” on page 11 for a description.	--with-pam=service name
TRACE_MSG_ BODY	Define if you want to include message bodies in trace information written with the *-t* or *-d* run-time option.	none

Qpopper Notes

- Qpopper uses the standard system authentication routines (which generally use `/etc/passwd` and `/etc/shadow`) to validate the user name in any mode (*user/pass*, *kerberos*, APOP) because the mail spool file must be owned by someone. The ownership relationships generally reside in `/etc/passwd`. Therefore, a user name must generally exist in both the *passwd* file as well as any of the other files associated with other authentication methods.
- When Qpopper is running, it moves your mail spool file to a file called `.user.pop` in the temporary spool location (mail spool directory is the default). `/tmp` can be an alternative location but is not recommended for security reasons. A system reboot will clear the files in `/tmp`. For performance reasons, a system administrator who has 1000+ users can create a separate spool directory, such as `/usr/spool/poptemp`, for Qpopper files. You can specify this directory to Qpopper by using the *--enable-temp-drop-dir* option when running `./configure`. Permissions should be the same as your mailspool with the same owner and group.
- Your spool directory needs to have ownership and permissions set correctly. Normally, this directory has owner `root` and group `mail`, and is set `drwxrwxr-x` or `drwxr-wxrwt`. (The second form sets the sticky bit to prevent non-owners from deleting or renaming files.)
- By default, Qpopper sets its group ID to the group of the spool directory before setting its UID to that of the user and giving up root privileges. Thus, by making the spool directory group `mail` and making it group writable, Qpopper is able to create and modify the user's spool and the dot-lock file. Since no users are part of the `mail` group, this is secure.
- **-no-mime**—As a way to enable MIME-mangling (reduce multipart MIME messages to a single text part, no attachments) with clients that do not support XMANGLE, add *-no-mime* to the user name. For example, if the user name is *mary*, enter it in the client as *mary-no-mime*.
- Do not use Qpopper over NFS. (But see *no-atomic-open* on page 35.)

Operating System-Specific

- **SCO**—Some versions of SCO use the *crypt_d* library, others the *crypt_i* library. This distribution assumes *crypt_d*. SCO requires loading the Standard and TCP/IP development environments to get the sockets and crypt libraries.

  Also, if you wish to use the elf binaries (to use dynamic instead of static linkage for C run-time functions), you need to edit the Makefiles so that the LIBS line is: *-lnsl -lsocket -lcrypt -lprot -lm -lx -ltinfo* and the CFLAGS line has *-s -O -belf*. (*-s* indicates strip the symbols (you can omit this if you choose), *-belf* indicates use elf binaries.)

- **IRIX**—The default spool directory is */usr/mail*; some systems use */usr/spool/mail*.

- **FreeBSD**—This requires the crypt library for password comparisons.

- **OSF/1**—If you are not using enhanced security (shadow passwords), then don't use *--enable-specialauth*. Otherwise, you receive a link error stating that *set_auth_parameters()* is not defined.

- **A/UX**— A/UX does not support the sticky bit, so the default directory is */tmp*. If you want to support shadow passwords, you need to use *--enable-specialauth* with*./configure*. A shadow password library is also required. You can find one on *jagubox.gsfc.nasa.gov*. A/UX requires GCC and libUTIL.a, also available on jagubox to build some versions of Qpopper.

- **NCR**—You may need to increase STRTHRESH, for example, a 600 user system needs to increase from 0x200000 to 0x600000.

- **NeXT**—You should probably use NetInfo Manager available under Next Admin to change your services file.

Managing Qpopper

Security and Authentication

In addition to the standard username and password, Qpopper can use APOP, Kerberos (version 4 or 5), or any PAM method. Qpopper can also use TLS/SSL to encrypt the authentication exchange.

Caution. To avoid the danger of sending clear text passwords over the network, use APOP, Kerberos, or TLS/SSL.

APOP

In APOP, the server issues a challenge, and the email client sends a response which proves it knows the password, without sending the actual password. Both the challenge and the response contain a random element, which prevents the response from being used by an interceptor.

APOP requires `dbm` or `gdbm` libraries to exist on the system. `gdbm` is a GNU's version of `dbm`, which can be obtained from any of GNU's distribution sites.

To setup APOP authentication, do the following:

1. Create a user account, for example POP, to be used for administering the APOP users.
2. Choose a location where `popauth` will place the authentication files, typically `/etc/pop.auth`.

   Caution. Make sure this is read/write accessible only to the administering account `pop`.
3. Run the configure script with the *--enable-*apop and *--with-popuid flags*, for example

   ```
   ./configure --enable-apop=/etc/pop.auth --enable-popuid=pop
   ```

   The first flag is the location of the authentication files; the second specifies the administering account that owns the authentication database.
4. Run **`make`**, this should produce executable files `popauth` and `popper`.
5. Move the executable files to a public location (normally you can run **`make install`** as `root` to do this).
6. Change the owner on `popauth` to the administering account (for example, `pop`) and set *suid*, for example:

   ```
   chown pop /usr/local/lib/popauth
   chmod u+s /usr/local/lib/popauth
   ```
7. Initialize the authentication database files by running the following command as `root`:
   `popauth -init`

8 New users can be added by `root` or the administering user (for example, `pop`) with the following command: **`popauth -user user`**

Or removed with the following command: **`popauth -delete user`**

Other users can add themselves or change their password with the following command: **`popauth`**

9 Scripts or other non-interactive processes can add or change the password for a user with the following command:

`popauth -user user password`

TLS/SSL Encryption

TLS/SSL allows all communications between Qpopper and an email client which supports TLS/SSL (such as Eudora) to be encrypted. This includes the contents of messages as well as authentications. TLS/SSL can keep confidential information or passwords from being intercepted in transit.

Note. TLS/SSL does not provide end-to-end encryption of email messages, not does it prevent or detect forged mail. But it can prevent passwords and mail messages from being seen by a network eavesdropper.

To use TLS/SSL, you must have `/dev/urandom` installed on your system. See your vendor if you need to obtain this.You also need TLS/SSL and cryptographic libraries, plus a security certificate and a public-private key pair. You also need to set TLS/SSL options, which indicate how you want Qpopper to support TLS/SSL (alternate port or STLS) and specify the certificate and private key.

If you use precompiled versions of Qpopper, you do not need to obtain TLS/SSL and cryptographic libraries. You can skip right to the section about getting your certificate.

Obtaining TLS/SSL and Cryptographic Libraries

Qpopper works with the TLS/SSL and cryptographic libraries in the free OpenSSL package (available at `<http://www.openssl.org>`) as well as SSL Plus and Security Builder® from Certicom (see <http://www.certicom.com>).

1 Obtain the libraries from either source

2 Install the libraries on your system (follow the instructions that came with the libraries)

3 Run `./configure`, adding *--with-openssl* if you installed the OpenSSL libraries, or *--with-sslplus* if you installed the SSL Plus libraries. (Add any other desired ./configure flags). For example:

`./configure --with-openssl --enable-specialauth --enable-timing`

Creating a Security Certificate

To create a certificate signed by a Certificate Authority using OpenSSL, follow these steps:

1. Create or choose a directory for the certificates and your private key. Because the private key is stored decrypted, it is very important that only user `root` has access to this directory. Assuming you choose `/etc/mail/certs` (which works as long as you do not have a user named `certs`), type the following three commands:

   ```
   mkdir -p -m665 /etc/mail/certs
   chown root:mail /etc/mail/certs
   chmod 660 /etc/mail/certs
   ```

2. Use `openssl` to create a public-private key pair and a certificate signing request (csr). For example, the following command (this text should be entered at a command prompt as one line):

   ```
   /usr/local/ssl/bin/openssl req -new -nodes -out req.pem -keyout
   /etc/mail/certs/cert.pem
   ```

   When you run `openssl` it prompts you for several items of information. It is very important that you properly answer these prompts; the default explanation in the prompt may not be accurate. It asks you:

   - **Country Name** — Supply the ISO-standard two-letter code for your country.
   - **State or Province Name** —Type the full name of your state or province.
   - **Locality Name** —Type the full name of your city or municipal area.
   - **Organization Name** — Type the legal name of your company or organization.
   - **Organizational Unit Name** — Type the name of your division or section of your company.
   - **Common Name** — Type the fully-qualified host name of the mail server host. Do not type your personal name, even if the `openssl` prompt directs you to type it. This must be the same name that a client enters to get to your server.
   - **Email Address** — This should be your email address, or that of an institutional role (such as `postmaster`).

3. Ensure that the file which now contains the private key (and will later contain the signed certificate) is owned by and only accessible by root. For example, the following two commands:

   ```
   chmod 600 /etc/mail/certs/cert.pem
   chown root:0 /etc/mail/certs/cert.pem
   ```

4. Send the certificate signing request (file `req.pem`) to your Certificate Authority for signing. You will receive back a signed request. Assuming this signed request is in a file called `signed_req.pem`, concatenate it to the private key generated earlier:

   ```
   cat signed_req.pem >> /etc/mail/certs/cert.pem
   ```

If you are using SSL Plus, see its documentation.

Setting TLS/SSL Options

After obtaining the TLS/SSL libraries, creating your private/public key pair, and obtaining your security certificate, you need to set certain TLS/SSL options. These options indicate how you want Qpopper to support TLS/SSL (alternate port or STLS), and specify the certificate and private key files.

1. Create a configuration file for Qpopper. You can locate this file anywhere you choose. For example, `/etc/mail/pop/qpopper.config`. Put the paths to the private key and signed certificate in this file, and enable either alternate-port or STLS. For example, using the above file names and STLS with OpenSSL:

   ```
   set tls-support = stls
   set tls-server-cert-file = /etc/mail/certs/cert.pem
   ```

   If you want to enable both alternate-port and STLS, you can do this with three configuration files. One file contains the `set tls-support = stls` command, and a second file contains the `set tls-support = alternate-port` command. Both files also contain one other command, which instructs Qpopper to read the third configuration file, for example, `set config-file = /etc/mail/pop/qpopper-tls.config`. Set all other options in this third file. Then use the *-f* run-time option to cause Qpopper to read either of the first two files.

   If you are using OpenSSL, you need to set *tls-server-cert-file*. If your private key is in a separate file, you also need to set *tls-private-key-file*. (In any case, the private key must not be encrypted.) **Caution:** Because your private key is in the clear, be certain the file is owned by `root` and no other users or groups have access.

   If you are using SSL Plus, you need to set *tls-identity-file* and *tls-passphrase*. With SSL Plus, your certificate and private key need to be in one file (called an *identity* file), and the private key must be encrypted. You specify the passphrase to decrypt the private key with the *tls-passphrase* option. **Caution:** Because your passphrase is in the clear in the configuration file, be certain that the configuration file is owned by `root` and no other users or groups have access.

2. Use the *-f config-file-path* command-line option to tell Qpopper to read the configuration file. For example, `-f /etc/mail/pop/qpopper-stls.config`.

PAM

PAM is an architecture which allows the use of various authentication modules with different applications. It is available on many platforms, including Linux and Solaris.

To use PAM, add the *--with-pam=service-name* flag when running `./configure`. If you omit *service-name* it defaults to `pop3`.

You must then create a file in `/etc/pam.d` with the same name as specified for *service-name*, for example, `/etc/pam.d/pop3`. This file contains the rules for authenticating using Qpopper. See your PAM documentation for more details.

An example of such a file is:

```
#%PAM-1.0
auth         required   /lib/security/pam_pwdb.so shadow
account      required   /lib/security/pam_pwdb.so
password     required   /lib/security/pam_cracklib.so
password    required  /lib/security/pam_pwdb.so nullok
                       use_authtok md5 shadow
session      required   /lib/security/pam_pwdb.so
```

Kerberos

Kerberos is a mechanism for secure authentication over untrustworthy networks. For more information, see the MIT Kerberos pages at <http://web.mit.edu/kerberos/www/>.

Caution. If you use Kerberos with Qpopper, be sure to obtain updated libraries that address CERT Advisory CA-2000-06 (see <http://www.cert.org/advisories/CA-2000-06.html>).

To use Kerberos with Qpopper, first obtain and install Kerberos libraries. We recommend using Kerberos version 5. To use Qpopper with Kerberos version 5, add the *--with-kerberos5* flag to your `./configure` command.

Using Bulletins

Bulletins can be used to send messages to all POP users. Bulletins are placed as plain text files in a specified format under a directory known to the server. Each bulletin has a name which starts with a unique, ascending number. Qpopper orders the bulletins by this number, and keeps track of which ones each user has seen. By default, new users get only the single most recent bulletin, but you can override this.

You can further classify bulletins by which group of users is to receive them.

Bulletins have two main advantages over simply sending an email to all users. First, the work of copying the bulletins is spread out over time, as each user checks mail. Second, new users get bulletins.

You must first enable bulletins, then write them.

Enabling Bulletins

To enable bulletins, do the following:

1. Choose the directory where the bulletins reside, usually `/var/spool/bulls`. The directory should be readable but not writable with user privileges, or make the permissions the same as the spool directory, for greater security.
2. Run `./configure` with the flag *--enable-bulletins* in addition to any other options. For example: **`./configure --enable-bulletins=/var/spool/bulls`**. Note that if you keep bulletins in `/var/spool/bulls`, you don't have to specify this, since it is the default location.
3. Run **`make`** and install as usual.
4. Use the command line option *-b* to override the compiled value for the bulletin directory, if desired.

If you want to use a central database instead of an individual file in the user's home directories to track which bulletins have been seen by which user, follow the instructions later in this section.

Writing Bulletins

To write a bulletin, do the following:

1. Create the bulletins as files. One easy way to do this is to use an email program (such as *Eudora*) and simply send yourself email, then do a *save as* to save the messages in files. You'll need to edit the files to add a "From " separator line, as described here.

2. Place the bulletins as files in the bulletins directory. The files should be readable, but not writable to everyone. You can chose readable file names for bulletins using the *number.string* form, for example, `00001.Bulletin_one`, `00002.four_hour_Downtime_2-4-98`, `00003.Quota-Revisions`.

3. Ensure that each bulletin starts with a unique and ascending order number. The number portion of each new filename should be one plus the largest bulletin number in the directory. (So if the current largest bulletin number is *00083*, use *00084* for the next bulletin.)

   Caution. You cannot recycle the bulletin numbers.

   You can limit bulletins to certain groups by using the *--enable-group-bulls* option with `./configure` or the *group-bulletins* configuration file option, and by inserting the group name as the second element in the bulletin file name.

   For example, the bulletin `001.staff.new_program_installed` would be seen only by members of the `staff` group.

4. Write the bulletin's header. Bulletins must be in the same format as messages in the mailspool. You must write the bulletin in the following format as shown in this example.

   ```
   From qpop Wed Nov  9 13:31:08 1994
   Date: Wed, 9 Nov 1994 13:31:07 -0800 (PST)
   To: user@localhost
   From: POP Administrator <postmaster@localhost>
   Subject: Example bulletin
   ```

   The first line must start with *From* and a space. It must be complete with address and date. An incorrect *From* line could cause the message to get concatenated to the previous message. Note that Qpopper replaces the *To:* line when the bulletin is copied to the user's mail spool.

5. Write the bulletin's message. There must an empty line (a line with just a new-line character, no spaces or tabs) between the header and message body, for example.

   ```
   From qpop Wed Nov  9 13:31:08 1994
   Date: Wed, 9 Nov 1994 13:31:07 -0800 (PST)
   To: user@localhost
   From: POP Administrator <postmaster@localhost>
   Subject: Example bulletin

   The system will be down for maintenance
   between 12 Midnight and 6 A.M. on November 30.
   ```

This bulletin is appended to the mailspool when the user checks his/her mail.

If you remove a file later on, it won't be seen by users who haven't checked their mail since you created the bulletin.

Note. Try to preserve the RFC 822 header format, especially date format. Failure to do so may cause email clients not to parse header information correctly.

Working with Bulletins

At the start of a POP session (after user authentication), Qpopper copies unread bulletins placed in the bulletins directory to the user's message spool. Qpopper figures out the last bulletin seen by the user by placing in the user's home directory a file called `.popbull`. This file contains the number of the last bulletin seen by the user. Any bulletin in the bulletins directory with a number greater than the one in `.popbull` is copied to the user's message spool.

A bulletin database can be used to track the bulletins instead of `.popbull` files in the users' home directory. This feature is enabled by using the *--enable-bulldb=/var/spool/bulls* option instead of the *--enable-bulletins* option with `./configure`.

If you use the bulletin database feature, it reads old `.popbull` files if they exist (but does not update them).

If you have a busy server, using a central database may impose concurrency issues. If too many users retrieve their mail at the same time, it may be difficult for some users to obtain write access to the bulletins database. By default, this results in an error.

If your system has the `usleep(3)` function, this is detected by the configure script, and should help the problem of multiple users. Even so, there is still a possibility that you may have a problem on an extremely busy server. If you want to use a bulletin database anyway (to avoid problems with users who lack a home directory or who exceed disk quota for their home directory), you can adjust the bulletin database behavior with run-time options.

Using the *-B* or *bulldb-nonfatal* option allows the POP session to continue even if the bulletins database can't be opened. As a result, your users will get their mail, but may not see some bulletins for possibly a long time, or even at all.

You can also set the maximum times Qpopper tries to lock the database with the *bulldb-max-retries* option. When the database is in use by another user, Qpopper tries repeatedly to access it, pausing for an amount of time between attempts. On systems with the `usleep(3)` function, this amount of time is a small random number of microseconds (somewhere between 1 microsecond and half a second), and the default value for *bulldb-max-retries* is 75. This usually results in a maximum delay well under a minute. On systems without the `usleep(3)` function, Qpopper waits between one and *bulldb-max-retries* seconds, which by default is 10. This may result in a maximum delay under two minutes, with far fewer attempts, and therefore less chance of success. To see if your system has the `usleep(3)` function, try **`man usleep`** or **`fgrep -i usleep config.h`** after running `./configure`. If you see `#define HAVE_USLEEP 1` then you have the `usleep(3)` function.

Configuring Bulletins for New Users

New users receive the single newest user bulletin by default. You can override this and specify how many bulletins new users should get (they will get that many of the newest bulletins). For example, pass *--enable-new-bulls=10* to `./configure` to give new users a maximum of ten bulletins, or add `set max-bulletins=10` to a configuration file.

Performance

Performance

Performance, that is, the hardware resources (such as CPU time and I/O bandwidth) needed to service a given number of users in a set amount of time, depends on many factors outside the scope of this manual. For example, your users' characteristics make a very big difference. Some users receive very little mail, check mail infrequently (perhaps every few days), and download and delete all their mail on each mail check. Other users receive massive amounts of email, keep all their mail on the server, and check mail constantly.

The type of hardware you use also has a very large impact. Generally speaking, SCSI or Firewire disk systems consume much less CPU per I/O and permit simultaneous I/O on separate channels. They are thus generally better suited for servers.

However, there are compile-time and run-time options which can affect Qpopper's performance.

In general, most administrators who are concerned about performance attempt to reduce the I/O needs of Qpopper.

The first option to consider for performance is *server mode*. Server mode reduces I/O in sessions in which all mail is deleted (the default with man email clients) or all mail is left on the server. It is safest when your users do not have shell access to the server. Server mode can be enabled for all users, all users who belong to a certain group, users who do not belong to a specified group, or on a user-by-user basis. See "Enabling Server Mode" on page 41 for more information.

Qpopper offers faster session start-ups when using server mode. In may cases, session start-ups with very large spools can be reduced to a few milliseconds from up to a minute (or even more).

Qpopper also offers the *fast-update* option which reduces I/O by a third during spool updates. **Caution:** use of this option is likely to break programs such as `biff(1)` or the shell's mail check feature. Only enable if such programs are not used. To be safe, don't set this option for users who have shell access to the system. This option is set with the *-F* command-line flag or the *fast-update* configure file option. Fast update is ideal for situations where a system acts only as a mail server.

In most UNIX/Linux systems, opening a file within a directory requires that the list of file names contained in the directory be read serially, until the desired file is found or all names have been read. This can result in a significant performance penalty when a large number of files exist in any directory. With email, this is most striking in the spool directory.

There are several things you can do to resolve this. A fairly easy method tells Qpopper to use a different directory for the temporary spool files it creates. This action alone can reduce the number of hits on your spool directory enough to make a difference. To use this method, decide on and create the new directory (make sure it has the same permissions

and ownership as the spool directory). Then use the *--enable-temp-drop-dir=path* flag with `./configure`, specifying the directory as *path*, or add `set temp-dir=`*`path`* to a configuration file.

You might also want to switch to hashed spool directories. Normally, the spool files for all users are located in the spool directory. With hashed spool directories, there is an extra layer or two of directories. This reduces the number of files in any one directory.

Qpopper supports the two most popular methods for hashed spool directories, called, imaginatively, *method 1* and *method 2* (or just *1* and *2*).

To enable hashed spool directories, use *--enable-hash-spool=1* or *--enable-hash-spool=2* with `./configure`, or add `set hash-spool=1` or `set hash-spool=2` to a configuration file.

When enabled, the subdirectory for a mail spool is determined from the user name by either hashing the first four characters (adding and then modulus 26) or by using directories equal to the first letter and the second letter (if any). For example, if the spool directory is `/var/mail`, the spool file for user `maida` would be:

`/var/mail/maida`	HASH_SPOOL not set
`/var/mail/o/maida`	method 1
`/var/mail/m/a/maida`	method 2

By default, when hashed spools are enabled, Qpopper checks whether or not the subdirectory or subdirectories exist for a user every time that user logs in. If the directory or directories do not exist, Qpopper creates them. This can be helpful when first moving to hashed spools but is unnecessary and a waste of time if the directories exist. Use the *--disable-hash-dir-check* flag with `./configure` or add `reset check-hash-spool` to a configuration file to prevent this.

Caution. If you disable this check, be sure to precreate all hashed spool subdirectories, or Qpopper (or the local delivery agent) may crash when trying to deliver mail to users whose containing directory does not yet exist.

Instead of hashed spool directories, some sites prefer to place mail spools in the user's home directory, typically in a file called `.mail`. This can avoid the problem of too many spools in one directory but does require that each user have a home directory. There may also be problems with users who exceed quota or have too many files in their home directory. This feature is called *home directory mail*, and is enabled by passing *--enable-home-dir-mail=file* to `./configure`. If you use the default file name of `.mail`, you don't need the *=file* part. You can also add `set home-dir-mail=`*`file`* to a configuration file.

Note that the *home-dir-mail* option only affects the spool file. If you want the `.`*`user`*`.pop` and the `.`*`user`*`.cache` files to be placed in the user's home directory as well, set the *home-dir-misc* option, either by passing *--enable-home-dir-misc* to `./configure`, or by adding `set home-dir-misc` to a configuration file.

When either hashed spool directories or home directory mail is used, by default Qpopper checks whether or not old temporary mail spool files exist in the spool directory. (This could happen if a very old version of Qpopper was used and was terminated before it could cleanup). This check is a waste of time in virtually all cases, but Qpopper is

designed to protect the safety of mail. You can turn off this check and save time by passing *--disable-old-spool-loc* to `./configure`, or adding reset `check-old-spool-loc` in a configuration file.

For example, your **`./configure`** command might look like:

```
./configure --enable-hash-spool=2 --disable-hash-dir-check
--disable-old-spool-loc --enable-specialauth
```

or, in a configuration file:

```
set hash-spool = 2

reset check-hash-spool
```

Adding `reset update-status-headers` to a configuration file, or use of the *--disable-status* `./configure` flag prevents Qpopper from creating or updating *Status:* and *X-UIDL:* headers. Combined with server mode, this further reduces I/O in sessions in which all mail is left on the server, and new mail has arrived. However, it also prevents Qpopper from keeping track of which messages have been downloaded, a feature relied upon by some email clients. In addition, it forces Qpopper to recalculate the unique identifier (UID) of each message, which increases CPU usage.

By default, Qpopper aggregates data to be sent to clients into large chunks. For example, an entire 1024-byte mail message might be sent in one packet. This may be faster or slower, depending on specifics of both the client and server hardware and networking stacks, as well as network elements in between (such as routers). Also, some networking stacks do their own aggregation.

Under congested network conditions, larger packets may increase the incidence of lost packets and thus client or server timeouts, leading to *POP timeout* or *EOF* errors.

When TSL/SSL is in effect, smaller packets increase the overhead needed to send data, which may result in worse performance.

You can adjust the Qpopper behavior by setting the *chunky-writes* configuration file option. See “chunky-writes” on page 30 for details.

Standalone mode may offer better performance than using `inetd`, but be aware that you may lose capabilities such as load throttling, address filtering, etc. Standalone mode is enabled with the *--enable-standalone* `./configure` flag. See “Enabling Standalone mode” on page 43 for more information.

Disabling reverse lookups avoids whatever overhead is incurred by the reverse DNS lookup. However, it may make your logs harder to read. Use the *-R* command-line switch or the *reverse-lookups* configuration file option.

You can adjust the frequency with which Qpopper calls kernel routines to check whether or not the mail lock needs to be refreshed during session start-up and shutdown. This is done with the *-L* command-line or *mail-lock-check* configuration file option; however, this option is probably not necessary in most cases. The option specifies the number of messages to be processed during initialization and cleanup before checking whether or not the mail lock needs refreshing. The default is 500. The value must be small enough to be processed in 60 seconds.

Troubleshooting

Troubleshooting

The first step in troubleshooting Qpopper is to try and `telnet` in to it. Generally, the easiest way to do this is to `telnet` from the host where you just installed Qpopper to itself. You need to specify the POP3 port in the `telnet` command. So, if you just installed Qpopper on a host called *penguin*, enter the following command: **`telnet penguin pop3`**.

`inetd` is not servicing the POP3 port if you receive one of the following error messages:

```
connect: Connection refused
connect: Connection closed
```

If you receive the first message, check your services file and make sure the port name *pop3* is exactly the same as the one in `inetd.conf`. Also, it can indicate that you have not reset `inetd` (**`kill -HUP inetd PID`**) (some systems can use **`inetd -c`**).

If you receive the second message, this indicates that `inetd` has the correct port assigned to Qpopper, but that either the program cannot be located, or it is failing on startup. Chances are Qpopper is not named correctly in the `/etc/inetd.conf` file.

Otherwise, add the *-d* option and check your log messages for the source of the problem. It may also be the case that you have duplicate port numbers assigned in `/etc/inetd.conf`. This is often indicated by a log entry containing the error message *address already in use*.

A tip to check that you have the Qpopper executable correctly listed in `inetd.conf` is to find the line in an editor, then copy the text specifying the executable path, then go to a command prompt and type **ls -l** and paste in the text. For example, in the `inetd.conf` line:

```
pop3    stream>tcp     root>/usr/local/lib/popper    qpopper -s
```

copy the underlined text.

If you are unsure where your log is located, try `/var/log/`, and if you still can't find it, see if you have a file called `/etc/syslog.conf`, and if so, look inside.

If you have correctly installed Qpopper as far as `inetd` is concerned, you will see the following line, which is the POP banner or greeting:

```
+OK QPOP (version 3.2) at host starting <13625.811191280@host>
```

Now, you need to enter two commands to authenticate yourself to Qpopper and gain access to your mail. Make sure you have a message or two queued so you can ensure that Qpopper is pointing at the correct mailspool file. Be aware that the password is echoed back:

you type: **`user your-user-name`**

you see: `+OK Password required for your-user-name`

you type: **`pass your-password`**

you see: `+OK` *`your-user-name`* `has 2 message(s) (4123 octets).`

You can enter **`quit`** to exit **`. list`** and **`uidl`** are two commands to list messages by size and unique identifier. At this point, Eudora or any other pop client should not have any problems communicating with your Qpopper.

If an error message displays indicating that your password is incorrect, you might not be using shadow passwords, and you may need to use the *--disable-specialauth* option when running `./configure` (after doing a **`make clean`**). Or, you might be using a user name whose UID is less than 11 which, by default, is automatically blocked from access. Normally, low UIDs are only used for `root` and daemon processes, which should not be checking mail.

Check the Qpopper Frequently Asked Questions (FAQ) page at `<http://www.qpopper.org/faq.html>` for more trouble-shooting tips. The Qpopper web page at `<http://www.qpopper.org>` has other resources that can also help.

Glossary

APOP. Authenticated Post Office Protocol is an MD5-based login command that does not send passwords in clear text over the network.

Boolean. Something that has two values, *true* and *false*.

Browser. A World Wide Web client that uses HTTP to read and display HTML documents.

CAPA. CAPA is a new POP extension (RFC 2449) which permits a POP server to give information to a client about other supported POP extensions, optional server behavior, and site policy.

Client. A computer or software program that accesses resources over the Internet. It is also an application that requests a server to perform a function. In the Internet mail environment, the term client indicates a mail user agent, for example, Eudora Pro.

Daemon. Daemons are generally server programs. They run continuously and are available when clients wish to initiate a session. However, an SMTP daemon periodically acts as a client when it needs to forward messages that are not to be delivered locally.

DBM. Database manager. It's a general term for a library that handles simple database functions, allowing a program such as Qpopper to store and access simple key/value pairs.

Domain. In general, a group of computers and other devices under the management of a single administrator or administrative entity. In the Internet, a domain identifies a range of IP addresses and mail-forwarding information. See DNS.

DNS. Domain Name Service. The naming service used by Internet Mail to support message routing. It maps domain addresses to IP addresses so Internet messages can be delivered to a particular server.

DRAC. Dynamic Relay Access Control. It's a third-party freeware package used to automate something called *POP before SMTP*, which means that users are permitted to relay messages through an SMTP server if they've checked mail using POP at the same site already. Over time it will be replaced by SMTP AUTH, which is the standards-based method and has other advantages.

Envelope. The envelope is the SMTP commands that enclose the email message. The SMTP envelope is the sequence of commands from the smtp-sender (a client) to the smtp-receiver (a server) to forward a message to the next hop in the delivery.

Finger query. A directory service that queries a Finger server.

FTP. File Transfer Protocol

Gateway. In general, software that translates information between one protocol and another.

GDBM. The GNU Database manager. It's free software from the GNU software people at <http://www.gnu.org>.

GNU. GNU's Not Unix. GNU is free software, everything except for the operating system itself (for that you use Linux or OpenBSD, FreeBSD, NetBSD, etc.). It's open-source software, so people all over the world contribute to it, and it's distributed for free. See <http://www.gnu.org>

Header. The part of an email message that precedes the body. It contains information such as the originator, recipient, and subject of the message. Also, it is used as a term for an individual header field, such as the *To:* header.

HTML. Hyper-Text Marking Language used to create Web pages. A Web client interprets HTML and displays documents and graphics accordingly. HTML also allows document authors to establish hypertext links between documents in various locations on the Internet and to create forms and image maps that enable users to interact with Web documents.

HTTP. Hyper-Text Transfer Protocol is the standard way of transferring information across the World Wide Web. It supports a variety of media and file formats across a variety of platforms.

IAB. Internet Architecture Board. The organization that sets high-level policy for Internet standards.

IETF. Internet Engineering Task Force. The standards-setting body of the Internet.

INETD. INETD is the standard Internet daemon. It is always running, and launches other programs, such as Qpopper, when a connection comes in from a client. You have to tell INETD which programs are for which ports, and how to run them.

Internet. A giant, global network made up of many smaller networks, all connected using the TCP/IP protocol. The Internet is the network of networks which spans the globe. TCP/IP is generally the network and transport protocol stack used to connect networks, but protocol translating gateways enable non-TCP/IP networks to connect to the Internet, as well.

IP address. The address that serves as a unique identifier of computers on the Internet. It is a sequence of 4 small integers (each less than 256). When written the numbers are separated by periods. For example:210.170.2.45. The DNS converts IP addresses to the more familiar domain names.

LDAP. Lightweight Directory Access Protocol is a protocol that provides an online, fully indexed, fast access white-pages directory service developed and freely distributed by the Regents of the University of Michigan.

Mailbox. See spool.

MIME. Multi-Purpose Internet Mail Extensions. A set of extensions to the Internet Mail standards that supports the inclusion of multilingual, multi-part and multimedia files, such as sound and video, in email messages.

Network. A group of connected computers that can communicate with one another. Networks enable computers to share files and resources and exchange messages.

NIS. Network Information Systems. It's a means of sharing information, such as user names and passwords across multiple systems, so that users can log in to any system, and the passwords are kept synchronized.

PAM. Pluggable Authentication Modules. It's a means of allowing any program such as Qpopper to authenticate with any mechanism, such as user/pass, LDAP, etc., by using PAM as the go-between. Qpopper asks PAM if the user is OK, and PAM asks the mechanisms it's been told to. Supported on many new UNIX/Linux systems.

Ph. Ph (Phone book) is a protocol providing an online, fully indexed, fast access white-pages directory service developed and freely distributed by the Computer and Communications Services Office at the University of Illinois at Urbana.

PID (Process ID). It's a unique number assigned to every active process. Once a process terminates its PID can be reassigned, but it's unique while the process is active.

POP3. Post Office Protocol 3. A protocol that provides a simple, standardized way for users to access mailboxes and download messages to their computers.

Postmaster. A special type of user responsible for maintaining the mail system for a particular group of computers. A postmaster is responsible for following up on queries from users and other postmasters. Internet standards require that the postmaster account be valid at every domain.

RFC. Request For Comments. In the Internet community, RFCs are a numbered sequence of documents generally describing protocols for Internet communication. An Internet standard protocol is also give a STD number, in addition to an RFC number. Only RFCs with a STD number are standards of the IETF. Some RFCs are historical or experimental, and are not standards. Others have not yet reached standard status. Still others provide documentation about the Internet itself.

Server. An entity that provides a network service. A server can be hardware (such as a file server), software (such as a mail server), or services (such as a transportation service). A mail server is a program that accepts, relays, and/or delivers mail.

SMTP. Simple Mail Transfer Protocol. The protocol widely implemented on the Internet for exchanging email messages.

Spool. A spool can refer to many different uses where information is held temporarily (for example, print spools). In this manual, spool refers to a mail spool. This is the location on a mail server where all mail for a user is deposited as it arrives. Mail is retrieved from the spool by a POP server (such as Qpopper) on request of a mail client (such as Eudora).

Static mail route. A mail route that an administrator explicitly specifies on a particular domain. Static routes bypass domain name systems (DNS).

TCP/IP. Transmission Control Protocol/Internet Protocol. TCP/IP is a set of protocols for computer network communication. The protocols provide conventions for connecting networks and routing traffic between them. It supports local area networks, as well as interconnections between local area networks. TCP/IP protocols are described in IETF RFCs, as well as numerous reference works.

UID. User Identifier. It's a unique number for every user name on a UNIX system. The system maps user names to user numbers internally.

UID. Unique Identifier. It's a unique number for every message in a mail spool. Clients rely on message UIDs staying the same.

UIDL. Unique Identifier List. See UID (Unique Identifier).

Username. A character string by which users are known (e.g., ldempster).

UUCP. UNIX to UNIX Copy Protocol is a UNIX email protocol.

World Wide Web. Also known as the Web, the World Wide Web is a graphical interface to Internet resources. Web refers to the set of hypermedia pages accessible via the Internet.