mpop

A POP3 client
version 1.0.28, 22 April 2013

Martin Lambers (marlam@marlam.de)

This manual was last updated 22 April 2013 for version 1.0.28 of mpop.

Table of Contents

1 Introduction

mpop is a POP3 client.

In its default mode of operation, it retrieves mails from one or more POP3 mailboxes, optionally does some filtering, and delivers them through a mail delivery agent (MDA) or to maildir folders, mbox files, or Exchange pickup directories. Mails that were successfully delivered before will not be retrieved a second time, even if errors occur or mpop is terminated in the middle of a session.

The best way to start is probably to have a look at the Examples section. See Chapter 11 [Examples], page 22.

In addition to the mail retrieval mode, mpop can be used in server information mode. In this mode, mpop prints as much information as it can get about a given POP3 server (greeting, supported features, login delay, maximum mail size, ...).

Normally, a configuration file contains information about which POP3 server to use and how to use it, but almost all settings can also be configured on the command line.

POP3 server information is organized in accounts. Each account describes one POP3 server: host name, authentication settings, TLS settings, and so on. Each configuration file can define multiple accounts.

Supported features include:

- Header based mail filtering: filter junk mail before downloading it
- Delivery to maildir folders, mbox files, Exchange pickup directories, or a mail delivery agent (MDA)
- Very fast POP3 implementation, using command pipelining
- Authentication methods USER/PASS, APOP, PLAIN, LOGIN and CRAM-MD5 (and GSSAPI, SCRAM-SHA-1, DIGEST-MD5, and NTLM when GNU SASL is used)
- TLS encrypted connections (including server certificate verification and the possibility to send a client certificate)
- Support for Internationalized Domain Names (IDN)
- IPv6 support
- support for multiple accounts

2 Configuration file

If it exists and is readable, a user configuration file will be loaded (`~/.mpoprc` by default). This file must have no more permissions than user read/write. Configuration file settings can be changed by command line options.

A configuration file is a simple text file. Empty lines and comment lines (whose first non-blank character is '#') are ignored. Every other line must contain a command and may contain an argument to that command. The argument may be enclosed in double quotes (").

If the first character of a filename is the tilde (`~`), this tilde will be replaced by `$HOME`.

If a command accepts the argument '`on`', it also accepts an empty argument and treats that as if it was '`on`'.

Commands form groups. Each group starts with the '`account`' command and defines the settings for one POP3 server.

See Chapter 11 [Examples], page 22.

2.1 General commands

'`defaults`'
: Set defaults. The following configuration commands will set default values for all following account definitions.

'`account` *name* `[ : ` *account*`[,...] ]`'
: Start a new account definition with the given name. The current default values are filled in (see [defaults], page 2).
If a colon and a list of previously defined accounts is given after the account name, the new account, with the filled in default values, will inherit all settings from the accounts in the list.

'`host` *hostname*'
: The POP3 server to retrieve mails from. The argument may be a host name or a network address. Every account definition must contain this command.

'`port` *number*'
: The port that the POP3 server listens on. The default is 110, unless TLS without STARTTLS is used, in which case it is 995.

'`timeout (off|`*seconds*`)`'
: Set or unset a network timeout, in seconds. The default is 180 seconds. The argument '`off`' means that no timeout will be set, which means that the operating system default will be used.

'`pipelining (auto|on|off)`'
: Enable or disable POP3 pipelining. The default is '`auto`', which means that mpop enables pipelining for POP3 servers that advertize this capability, and disables it for all other servers. See Chapter 6 [Pipelining], page 16.

'`received_header [(on|off)]`'
: Enable or disable the Received header. By default, mpop prepends a Received header to the mail during delivery. This is required by the RFCs if the mail is

subsequently further delivered e.g. via SMTP, and it is a good idea in all other cases. Nevertheless, if you absolutely have to, you can disable the Received header with this command.

2.2 Authentication commands

See Chapter 5 [Authentication], page 14.

'`auth [(on|method)]`'
: This command chooses the POP3 authentication method. With the argument '`on`', mpop will choose the best one available for you (see below). This is the default. Accepted methods are '`user`', '`apop`', '`plain`', '`scram-sha-1`', '`cram-md5`', '`gssapi`', '`digest-md5`', '`external`', '`login`', and '`ntlm`'. See Chapter 5 [Authentication], page 14.

'`user [username]`'
: Set your user name for POP3 authentication. An empty argument unsets the user name.

'`password [secret]`'
: Set your password for POP3 authentication. An empty argument unsets the password. If no password is set but one is needed during authentication, mpop will try to find it. First, if '`passwordeval`' is set, it will evaluate that command. If '`passwordeval`' is not set, mpop will try to find the password in `~/.netrc`. If that fails, it will try to find it in `SYSCONFDIR/netrc` (use `--version` to find out what `SYSCONFDIR` is on your platform). If that fails, it will try to get it from a system specific keyring (if available). If that fails, mpop will prompt you for it. See Chapter 5 [Authentication], page 14.

'`passwordeval [eval]`'
: Set your password for authentication to the output (stdout) of the execution of *eval*.

'`ntlmdomain [ntlmdomain]`'
: Set a domain for the '`ntlm`' authentication method. The default is to use no domain (equivalent to an empty argument), but some servers seem to require one, even if it is an arbitrary string.

2.3 TLS commands

See Chapter 4 [Transport Layer Security], page 12.

'`tls [(on|off)]`'
: This command enables or disables TLS/SSL encrypted connections to the POP3 server. Not every server supports TLS, and many that support it require the '`tls_starttls off`' command.
To use TLS/SSL, it is required to either use the '`tls_trust_file`' command (highly recommended) or to disable '`tls_certcheck`'. See Chapter 4 [Transport Layer Security], page 12.

‘`tls_starttls [(on|off)]`’
: This command chooses the TLS/SSL variant: with STARTTLS (‘`on`’, default) or POP3-over-TLS (‘`off`’). Most servers support the latter variant, which is also commonly referred to as `"POP3 with SSL"`. See Chapter 4 [Transport Layer Security], page 12.

‘`tls_trust_file [file]`’
: This command activates strict server certificate verification. The given file must contain one or more certificates of trusted Certification Authorities (CAs) in PEM format.
On Debian based systems, you can install the ‘`ca-certificates`’ package and use the file ‘`/etc/ssl/certs/ca-certificates.crt`’.
An empty argument disables this feature. See Chapter 4 [Transport Layer Security], page 12.

‘`tls_crl_file [file]`’
: This command sets or unsets a certificate revocation list (CRL) file for TLS, to be used during strict server certificate verification as enabled by the [tls_trust_file], page 4 command. This allows the verification procedure to detect revoked certificates. See Chapter 4 [Transport Layer Security], page 12.

‘`tls_fingerprint [fingerprint]`’
: This command sets or unsets the fingerprint of a particular TLS certificate. This certificate will then be trusted, regardless of its contents. This can be used to trust broken certificates (e.g. with a non-matching hostname) or in situations where ‘`tls_trust_file`’ cannot be used for some reason. You can give either an SHA1 (recommended) or an MD5 fingerprint in the format `01:23:45:67:...`. You can use ‘`--serverinfo --tls --tls-certcheck=off`’ to get the peer certificate’s fingerprints. See Chapter 4 [Transport Layer Security], page 12.

‘`tls_key_file [file]`’
: This command (together with the ‘`tls_cert_file`’) command enables mpop to send a client certificate to the POP3 server if requested. The file must contain the private key of a certificate in PEM format. An empty argument disables this feature. See Chapter 4 [Transport Layer Security], page 12.

‘`tls_cert_file [file]`’
: This command (together with the ‘`tls_key_file`’ command) enables mpop to send a client certificate to the POP3 server if requested. The file must contain a certificate in PEM format. An empty argument disables this feature. See Chapter 4 [Transport Layer Security], page 12.

‘`tls_certcheck [(on|off)]`’
: This command enables or disables checks for the server certificate.
WARNING: When the checks are disabled, TLS/SSL sessions will be vulnerable to man-in-the-middle attacks! See Chapter 4 [Transport Layer Security], page 12.

‘`tls_force_sslv3 [(on|off)]`’
: Force TLS/SSL version SSLv3. This might be needed to use SSL with some old and broken servers. Do not use this unless you have to. See Chapter 4 [Transport Layer Security], page 12.

‘`tls_min_dh_prime_bits [bits]`’
: Set or unset the minimum number of Diffie-Hellman (DH) prime bits that mpop will accept for TLS sessions. The default is set by the TLS library and can be selected by using an empty argument to this command. Only lower the default (for example to 512 bits) if there is no other way to make TLS work with the remote server. See Chapter 4 [Transport Layer Security], page 12.

‘`tls_priorities [priorities]`’
: Set the priorities for TLS sessions. The default is set by the TLS library and can be selected by using an empty argument to this command. Currently this command only works with sufficiently recent GnuTLS releases. See the GnuTLS documentation of the ‘`gnutls_priority_init`’ function for a description of the *priorities* string. See Chapter 4 [Transport Layer Security], page 12.

2.4 Commands specific to mail retrieval mode

See Chapter 8 [Mail retrieval mode], page 18.

‘`delivery method method_arguments...`’
: How to deliver messages received from this account.

- delivery mda *command*
  Deliver the mails through a mail delivery agent (MDA).
  All occurences of `%F` in the command will be replaced with the envelope from address of the current message (or MAILER-DAEMON if none is found). Note that this address is guaranteed to contain only letters `a-z` and `A-Z`, digits `0-9`, and any of `.@_-+/`, even though that is only a subset of what is theoretically allowed in a mail address. Other characters, including those interpreted by the shell, are replaced with `_`. Nevertheless, you should put `%F` into single quotes: `'%F'`.
  Use `delivery mda "/usr/bin/procmail -f '%F' -d $USER"` for the procmail MDA.
  Use `delivery mda "/usr/sbin/sendmail -oi -oem -f '%F' -- $USER"` to let your MTA handle the mail.
  Use `delivery mda /usr/local/bin/msmtp --host=localhost --from='%F' -- $USER@‘hostname‘.‘dnsdomainname‘"` to pass the mail to your MTA via SMTP. (This is what fetchmail does by default.)
- delivery maildir *directory*
  Deliver the mails to the given maildir directory. The directory must exist and it must be a valid maildir directory; mpop will not create directories. This delivery type only works on file systems that support hard links.
- delivery mbox *mbox-file*
  Deliver the mails to the given file in mbox format. The file will be locked with `fcntl(2)`. mpop uses the MBOXRD mbox format variant; see the documentation of the mbox format.

- delivery exchange *directory*
  Deliver the mails to the given Exchange pickup directory. The directory must exist.

If the delivery method needs to parse the mail headers for an envelope from address (the mda method if the command contains `%F`, and the mbox method), then it needs to create a temporary file to store the mail headers (but not the body) in. See `$TMPDIR` in [Environment / Files], page 8.

'`uidls_file` *`filename`*'
: The file to store UIDLs in. These are needed to identify new messages. `%U` in the filename will be replaced by the username of the current account. `%H` in the filename will be replaced by the hostname of the current account. If the filename contains directories that do not exist, mpop will create them. mpop locks this file for exclusive access when accessing the associated POP3 account. The default value is `~/.mpop_uidls/%U_at_%H`. You can also use a single UIDLS file for multiple accounts, but then you cannot poll more than one of these accounts at the same time.

'`only_new [(on|off)]`'
: By default, mpop processes only new messages (new messages are those that were not already successfully retrieved in an earlier session). If this option is turned off, mpop will process all messages.

'`keep [(on|off)]`'
: Keep all mails on the POP3 server, never delete them. The default behavior is to delete mails that have been successfully delivered or filtered by kill filters.

'`killsize (off|`*`size`*`)`'
: Mails larger than the given size will be deleted, not downloaded (unless the keep command is used, in which case they will just be skipped). The size argument must be zero or greater. If it is followed by a 'k' or an 'm', the size is measured in kibibytes/mebibytes instead of bytes. Note that some POP3 servers report slightly incorrect sizes for mails. See Chapter 10 [Filtering], page 21.
  When '`killsize`' is set to 0 and '`keep`' is set to on, then all mails are marked as retrieved, but no mail gets deleted from the server. This can be used to synchronize the UID list on the client to the UID list on the server.

'`skipsize (off|`*`size`*`)`'
: Mails larger than the given size will be skipped (not downloaded). The size argument must be zero or greater. If it is followed by a 'k' or an 'm', the size is measured in kibibytes/mebibytes instead of bytes. Note that some POP3 servers report slightly incorrect sizes for mails. See Chapter 10 [Filtering], page 21.

'`filter [`*`COMMAND`*`]`'
: Set a filter which will decide whether to retrieve, skip, or delete each mail by investigating the mail's headers. The POP3 server must support the POP3 TOP command for this to work; see Chapter 9 [Server information mode], page 19. An empty argument disables filtering.
  All occurences of `%F` in the command will be replaced with the envelope from

address of the current message (or MAILER-DAEMON if none is found). Note that this address is guaranteed to contain only letters `a-z` and `A-Z`, digits `0-9`, and any of `.@_-+/`, even though that is only a subset of what is theoretically allowed in a mail address. Other characters, including those interpreted by the shell, are replaced with `_`. Nevertheless, you should put `%F` into single quotes: `'%F'`.
All occurences of `%S` in the command will be replaced with the size of the current mail as reported by the POP3 server.
The mail headers (plus the blank line separating the headers from the body) will be piped to the command. Based on the return code, mpop decides what to do with the mail:

- 0: proceed normally; no special action
- 1: delete the mail; do not retrieve it
- 2: skip the mail; do not retrieve it

Return codes greater than or equal to 3 mean that an error occurred. The `sysexits.h` error codes may be used to give information about the kind of the error, but this is not necessary. See Chapter 10 [Filtering], page 21.

3 Invocation

3.1 Synopsis

- Mail retrieval mode (default):
  `mpop [`*option*`...] [--] [`*account*`...]`

- Server information mode:
  `mpop [`*option*`...] --serverinfo [`*account*`...]`

3.2 Exit code

The standard exit codes from `sysexits.h` are used.

3.3 Environment / Files

'`~/.mpoprc`'
: The default user configuration file.

'`~/.mpop_uidls`'
: Default directory to store UIDLs files in.

'`~/.netrc and SYSCONFDIR/netrc`'
: The `netrc` file contains login information. If a password is not found in the configuration file, mpop will search it in `~/.netrc` and `SYSCONFDIR` before prompting the user for it. The syntax of `netrc` files is described in the `netrc(5)` or `ftp(1)` manual page.

'`$USER, $LOGNAME`'
: These variables override the user's login name. `$LOGNAME` is only used if `$USER` is unset. The user's login name is used for `Received` headers.

'`$TMPDIR`' Directory to create temporary files in. If this is unset, a system specific default directory is used.

3.4 Options

Options override configuration file settings. The following options are accepted:

3.4.1 General options

'`--version`'
: Print version information. This includes information about the library used for TLS/SSL support (if any), the library used for authentication, and the authentication mechanisms supported by this library.

'`--help`' Print help.

‘`-P`’
‘`--pretend`’
: Print the configuration settings that would be used, but do not take further action. An asterisk (’*’) will be printed instead of the password.

‘`-d`’
‘`--debug`’
: Print lots of debugging information, including the whole conversation with the POP3 server. Be careful with this option: the (potentially dangerous) output will not be sanitized, and your password may get printed in an easily decodable format!
This option implies ‘`--half-quiet`’, because the debugging output would otherwise interfere with the progress output.

3.4.2 Changing the mode of operation

‘`-S`’
‘`--serverinfo`’
: Print information about the POP3 server and exit. This includes information about supported features (pipelining, authentication methods, TOP command, . . .), about parameters (time for which mails will not be deleted, minimum time between logins, . . .), and about the TLS certificate (if TLS is active). See Chapter 9 [Server information mode], page 19.

3.4.3 Configuration options

Most options in this category correspond to a configuration file command. Please refer to Chapter 2 [Configuration file], page 2 for detailed information.

‘`-C` *`filename`*’
‘`--file=`*`filename`*’
: Use the given file instead of `~/.mpoprc` as the configuration file.

‘`--host=`*`hostname`*’
: Use this POP3 server with settings from the command line; do not use any configuration file data. This option disables loading of the configuration file. You cannot use both this option and account names on the command line.

‘`--port=`*`number`*’
: Set the port number to connect to. See [port], page 2.

‘`--timeout=(off|`*`seconds`*`)`’
: Set or unset a network timeout, in seconds. See [timeout], page 2.

‘`--pipelining=(auto|on|off)`’
: Enable or disable POP3 pipelining. See [pipelining], page 2.

‘`--received-header[=(on|off)]`’
: Enable or disable the Received header. See [received_header], page 2.

‘`--auth[=(on|`*`method`*`)]`’
: Set the authentication method to automatic (with "on") or manually choose an authentication method. See [auth], page 3.

‘`--user=[username]`’
: Set or unset the user name for authentication. See [user], page 3.

‘`--passwordeval=[eval]`’
: Evaluate password for authentication. See [passwordeval], page 3.

‘`--tls[=(on|off)]`’
: Enable or disable TLS/SSL. See [tls], page 3.

‘`--tls-starttls[=(on|off)]`’
: Enable or disable STARTTLS for TLS encryption. See [tls_starttls], page 3.

‘`--tls-trust-file=[file]`’
: Set or unset a trust file for TLS encryption. See [tls_trust_file], page 4.

‘`--tls-crl-file=[file]`’
: Set or unset a certificate revocation list (CRL) file for TLS. See [tls_crl_file], page 4.

‘`--tls-fingerprint=[fingerprint]`’
: Set ot unset the fingerprint of a trusted TLS certificate. See [tls_fingerprint], page 4.

‘`--tls-key-file=[file]`’
: Set or unset a key file for TLS encryption. See [tls_key_file], page 4.

‘`--tls-cert-file=[file]`’
: Set or unset a cert file for TLS encryption. See [tls_cert_file], page 4.

‘`--tls-certcheck[=(on|off)]`’
: Enable or disable server certificate checks for TLS encryption. See [tls_certcheck], page 4.

‘`--tls-force-sslv3=(on|off)]`’
: Force TLS/SSL version SSLv3. See [tls_force_sslv3], page 4.

‘`--tls-min-dh-prime-bits=[bits]`’
: Set or unset minimum bit size of the Diffie-Hellman (DH) prime. See [tls_min_dh_prime_bits], page 5.

‘`--tls-priorities=[priorities]`’
: Set or unset TLS priorities. See [tls_priorities], page 5.

3.4.4 Options specific to mail retrieval mode

‘`-q`’
‘`--quiet`’ Do not print status or progress information.

‘`-Q`’
‘`--half-quiet`’
: Print status but not progress information.

‘`-a`’
‘`--all-accounts`’
: Query all accounts in the configuration file.

‘`-A`’
‘`--auth-only`’
: Authenticate only; do not retrieve mail. Useful for SMTP-after-POP.

‘`-s`’
‘`--status-only`’
: Print number and size of mails in each account only; do not retrieve mail.

‘`-n`’
‘`--only-new[=(on|off)]`’
: Process only new messages. See [only_new], page 6.

‘`-k`’
‘`--keep[=(on|off)]`’
: Do not delete mails from POP3 servers, regardless of other options or settings. See [keep], page 6.

‘`--killsize=(off|`*`size`*`)`’
: Set or unset kill size. See [killsize], page 6.

‘`--skipsize=(off|`*`size`*`)`’
: Set or unset skip size. See [skipsize], page 6.

‘`--filter=[`*`command`*`]`’
: Set a filter which will decide whether to retrieve, skip, or delete each mail by investigating the mail's headers. See [filter], page 6.

‘`--delivery=`*`method,method_arguments...`*’
: How to deliver messages received from this account. See [delivery], page 5. Note that a comma is used instead of a blank to separate the method from its arguments.

‘`--uidls-file=`*`filename`*’
: File to store UIDLs in. See [uidls_file], page 6.

4 Transport Layer Security

Transport Layer Security (TLS) is a new name for Secure Socket Layer (SSL). The TLS 1.0 protocol is an updated version of the SSL 3.0 protocol. TLS and SSL mean the same thing.

Quoting from RFC2246 - the TLS 1.0 protocol specification:
"The TLS protocol provides communications privacy over the Internet. The protocol allows client/server applications to communicate in a way that is designed to prevent eavesdropping, tampering, or message forgery."

POP3 servers can use TLS in one of two modes:

- Immediately
  This is known as POP3 tunneled through TLS. The default port for this mode is 995 (pop3s). This is what most servers support, and is often simply called "POP3 with SSL".
- Via the STARTTLS POP3 command
  The POP3 session begins normally. The client sends the STLS command when it wishes to begin TLS encryption. The default port for this mode is the default POP3 port: 110 (pop3).

mpop can switch between these modes with the '`tls_starttls`' command (see [tls_starttls], page 3) command or the '`--tls-starttls`' option (see [–tls-starttls], page 10).

When TLS is started, the server sends a certificate to identify itself. This certificate contains information about the certificate owner, the certificate issuer, and the activation and expiration times of the certificate. This information can be displayed in server information mode. See Chapter 9 [Server information mode], page 19.

To use TLS, it is required to either enable full server certificate verification using the '`tls_trust_file`' command or '`--tls-trust-file`' option, or to trust one particular peer certificate using the '`tls_fingerprint`' command or '`--tls-fingerprint`' option, or to disable all certificate checks using '`tls_certcheck off`' or '`--tls-certcheck=off`'. WARNING: When certificate checks are disabled, TLS/SSL sessions are vulnerable to man-in-the-middle attacks! See [tls_trust_file], page 4, [–tls-trust-file], page 10, [tls_fingerprint], page 4, [–tls-fingerprint], page 10, [tls_certcheck], page 4, [–tls-certcheck], page 10.

If your system has a file that collects all system-wide trusted CA certificates, it is easiest to just use this in the '`defaults`' section of your configuration file. On Debian-based systems, for example, the adequate command would be '`tls_trust_file /etc/ssl/certs/ca-certificates.crt`'.

But you can also find out manually which CA certificate you need to trust. First, issue the following command:

```
$ mpop --serverinfo --host=pop.example.com --tls=on --tls-certcheck=off
```

The option '`--tls-certcheck=off`' allows mpop to accept any certificate, so that it can print some information about it. The output of this command tells you the common name of the server certificate issuer. You have to trust this issuer to use full TLS security. Usually you can find the CA certificate on the issuer's homepage. With this CA certificate, the following should succeed:

```
$ mpop --serverinfo --host=pop.example.com --tls=on \
  --tls-trust-file=ca_cert.txt
```

If the server requests it, the client can send a certificate, too. This allows the server to verify the identity of the client. See the EXTERNAL mechanism in Chapter 5 [Authentication], page 14. The '`tls_key_file`'/'`tls_cert_file`' commands or the '`--tls-key-file`'/'`--tls-cert-file`' options can be used to set a client certificate. See [tls_key_file], page 4/[–tls-key-file], page 10, [tls_cert_file], page 4/[–tls-cert-file], page 10. Note that GnuTLS will only send a client certificate if it matches one of the CAs advertised by the server. If you set a client certificate but it is not sent to the server, probably does not match any CA that the server trusts.

If you need to fine tune TLS parameters or have problems connecting to your server, have a look at the [tls_force_sslv3], page 4, [tls_min_dh_prime_bits], page 5, and [tls_priorities], page 5 commands.

5 Authentication

POP3 servers require a client to authenticate itself before it is allowed to retrieve mail.

Multiple authentication methods exist. Most POP3 servers support only some of them. Some methods send authentication data in plain text (or nearly plain text) to the server, and some others are vulnerable to attacks. These methods should only be used when TLS is active to prevent others from stealing the password. See Chapter 4 [Transport Layer Security], page 12.

By default, mpop chooses a method automatically, and it will never choose one that puts the authentication data at risk. See below for details.

mpop supports the following authentication methods:

- 'USER'
  This authentication method needs a user name and a password. Both are send in plain text. All POP3 servers support this authentication method.
- 'PLAIN'
  This authentication method needs a user name and a password. Both are send in BASE64 encoding, which can be easily decoded to plain text.
- 'APOP'
  This authentication method needs a user name and a password. The authentication data is not sent in plain text, but APOP is vulnerable to man-in-the-middle attacks. This method should not be used unless TLS is active.
- 'SCRAM-SHA-1'
  This authentication method needs a user name and a password. The authentication data is not sent in plain text, which means this method can safely be used without TLS.
- 'CRAM-MD5'
  This authentication method needs a user name and a password. The authentication data is not sent in plain text, which means this method can safely be used without TLS.
- 'GSSAPI'
  This authentication method needs a user name. The Kerberos framework takes care of secure authentication, therefore this method can safely be used without TLS.
- 'EXTERNAL'
  This is a special authentication method: The actual authentication happens outside of the POP3 protocol, typically by sending a TLS client certificate (see Chapter 4 [Transport Layer Security], page 12).
  The EXTERNAL method merely confirms that this authentication succeeded for the given user (or, if no user name is given, confirms that authentication succeeded). Thus it may not be necessary for authentication to use this method, and if the server does not support the EXTERNAL method, this does not mean that it does not support authentication with TLS client certificates.
  This authentication method is not chosen automatically; you have to request it manually.
  Note: (SMTP) Sendmail 8.12.11 advertises the EXTERNAL mechanism only after a

TLS client certificate has been send. It seems to ignore the optional user name. Does anyone know more about this?

- ‘`DIGEST-MD5`’
  This is an obsolete authentication method needs a user name and a password. The authentication data is not sent in plain text, but the encryption based on MD5 is not considered secure anymore.
- ‘`LOGIN`’
  This is a non-standard authentication method similar to (but worse than) PLAIN. It needs a user name and a password, both of which are send in BASE64 encoding, which can be easily decoded to plain text.
- ‘`NTLM`’
  This is an obscure non-standard authentication method. It needs a user name and a password and in some cases a special domain parameter (see [ntlmdomain], page 3). The authentication data is not send in plain text, but since NTLM is not an open standard, it should be considered broken and insecure.

It depends on the underlying authentication library and its version whether a particular method is supported or not. Use ‘`--version`’ to find out which methods are supported by your version of mpop.

Authentication data can be set with the ‘`user`’ and ‘`password`’ commands or with the ‘`--user`’ option. See [user], page 3, [password], page 3, [–user], page 9. If no password is set but one is needed during authentication, mpop will try to find it. First, if ‘`passwordeval`’ is set, it will evaluate that command. If ‘`passwordeval`’ is not set, mpop will try to find the password in `~/.netrc`. If that fails, it will try to find it in `SYSCONFDIR/netrc` (use `--version` to find out what `SYSCONFDIR` is on your platform). If that fails, it will try to get it from a system specific keyring (if available). If that fails, mpop will prompt you for it.

Currently supported keyrings are the Gnome Keyring and the Mac OS X Keychain. The script `mpop-gnome-tool.py` can be used to manage Gnome Keyring passwords for mpop. To manage Mac OS X Keychain passwords, use the Keychain Access GUI application. The ‘`account name`’ is same as the mpop ‘`user`’ argument. The ‘`keychain item name`’ is `pop3://<hostname>` where `<hostname>` matches the mpop ‘`host`’ argument.

The authentication method can be chosen with the ‘`auth`’ command or ‘`--auth`’ option, but it is usually sufficient to just use the ‘`on`’ argument to let mpop choose the method itself. See [auth], page 3, [–auth], page 9.

If mpop chooses the method itself, it will never choose an insecure method. If TLS is active, all methods are considered secure in this context, because the connection to the server is protected by TLS. If TLS is not active, only the SCRAM-SHA-1, CRAM-MD5, and GSSAPI methods are considered secure in this context, because all the others methods put the authentication data at risk.

If you really want to risk your authentication data, you have to force mpop to do that by manually setting the authentication method while TLS is off.

6 Pipelining

A POP3 client that sends multiple POP3 commands at once to a POP3 server before starting to read the server's answers is using POP3 pipelining. Since the client does not have to wait for the server's answer before sending the next command, and the server does not have to wait for the next command from the client, pipelining can speed up a POP3 session substantially.

Pipelining in mpop works by sending up to '`PIPELINE_MAX`' commands to the server, then begin to read its answers, and refill the command pipeline when the number of unanswered commands drops to '`PIPELINE_MIN`'. '`PIPELINE_MIN`' and '`PIPELINE_MAX`' are compile time constants.

By default, mpop will enable pipelining automatically if the server supports the CAPA command and advertizes the pipelining capability, and disable it for all other servers. See Chapter 9 [Server information mode], page 19.

You can change this behaviour with the '`pipelining`' command or '`--pipelining`' option. See [pipelining], page 2, [–pipelining], page 9. It is always safe to disable pipelining. It is not recommended to force pipelining for servers that are not known to support it.

7 Defective POP3 servers

Some POP3 servers still do not support the UIDL command. In this case, mpop cannot recognize messages that were already successfully retrieved, and will treat all messages as new. Use the '`--serverinfo`' option to find out if a server supports the UIDL command.

Some POP3 servers count end-of-line characters as two bytes (CRLF) instead of one (LF), so that the size of a mail as reported by the POP3 server is slightly larger than the actual size. This has the following consequences: The size filters are not accurate. Do not rely on exact size filtering. The progress output may display inaccurate (slightly too low) percentage values for the first mail retrieved from a POP3 server. mpop will detect this after the first mail has been read and will display corrected values for subsequent mails.

8 Mail retrieval mode

In this mode, mpop retrieves mail from one or more POP3 servers. It delivers each of them using the method that was given with the '`delivery`' command or '`--delivery`' option. See [delivery], page 5, [–delivery], page 11.

While retrieving the mail, mpop displays approximate progress information, which can be turned off with the '`--half-quiet`' or '`--quiet`' options; see [–half-quiet], page 10, [–quiet], page 10.

If the delivery succeeded, the mail is deleted from the POP3 server by default. The '`keep`' command and '`--keep`' option can prevent the deletion of mails; see [keep], page 6, [–keep], page 11.

Important note: Some POP3 servers will delete mails without any user interaction. See EXPIRE in Chapter 9 [Server information mode], page 19. mpop can do nothing about that.

If you don't want to download certain mails, but skip them or delete them directly, you can do filtering based on the mail headers. See Chapter 10 [Filtering], page 21.

If you just want to know if you have new mails (and how many), use the '`--status-only`' option. See [–status-only], page 11.

If you just want to authenticate to the POP3 server, but don't want to look at your mails, use the '`--auth-only`' option. See [–auth-only], page 10. This can be useful for sending mail through SMTP servers that require SMTP-after-POP (aka POP-before-SMTP).

Before mpop delivers a mail, it prepends a Received header to it. This is necessary if the delivery method transmits the mail to an SMTP server, for example. mpop does not change the contents of the mail in any other way.

9 Server information mode

In server information mode, mpop prints as much information about the POP3 server as it can get and then exits.

The POP3 features that can be detected are:

- IMPLEMENTATION
  The implementation string of the POP3 server.
- CAPA
  Support for the POP3 CAPA command. The server sends a list of its capabilities in response to this command.
- PIPELINING
  Support for POP3 pipelining. See Chapter 6 [Pipelining], page 16.
- TOP
  Support for the POP3 TOP command. This is needed for header based filtering to work. See Chapter 10 [Filtering], page 21.
- UIDL
  Support for the POP3 UIDL command. This is needed to distinguish between new and already retrieved messages.
- LOGIN-DELAY
  The minimum time between two POP3 sessions. The server may refuse a POP3 session if the last one was active less than this time period ago.
- EXPIRE
  The time after which old mails are deleted by the POP3 server.
  - NEVER: The POP3 server will not delete mail without the user requesting it.
  - 0: The POP3 server will not keep mails; all mails will be deleted after they have been downloaded, regardless of the user's wishes.
  - *number*: The number of days that the POP3 server will keep mails before deleting them without user interaction.
- STARTTLS
  See Chapter 4 [Transport Layer Security], page 12.
- AUTH
  See Chapter 5 [Authentication], page 14.
- RESP-CODES
  If authentication fails and the POP3 server issues an error message beginning with a square bracket, this message will include additional information about the source of the error:
  - `[LOGIN-DELAY]`: The login delay period hast not yet expired.
  - `[IN-USE]`: Authentication succeeded but the mailbox is currently in use, possibly by another POP3 session.
- AUTH-RESP-CODE
  If authentication fails and the POP3 server issues an error message beginning with a square bracket, this message will include additional information about the source of the error:

- `[LOGIN-DELAY]`: The login delay period hast not yet expired.
- `[IN-USE]`: Authentication succeeded but the mailbox is currently in use, possibly by another POP3 session.
- `[SYS/TEMP]`: Temporary system failure; try again later.
- `[SYS/PERM]`: Permanent system failure; ask the administrator.
- `[AUTH]`: Incorrect user name or password or some other problem with the user's credentials.

If TLS is activated for server information mode, the following information will be printed about the POP3 server's TLS certificate (if available):

- Owner information
  - Common Name
  - Organization
  - Organizational unit
  - Locality
  - State or Province
  - Country
- Issuer information
  - Common Name
  - Organization
  - Organizational unit
  - Locality
  - State or Province
  - Country
- General
  - Activation time
  - Expiration time
  - SHA1 fingerprint
  - MD5 fingerprint

10 Filtering

There are three filtering commands available. They will be executed in the following order:

1. '`killsize`'
2. '`skipsize`'
3. '`filter`'

If a filtering command applies to a mail, the remaining filters will not be executed.

The POP3 server must support the POP3 TOP command (Chapter 9 [Server information mode], page 19) for filtering with a filter command: It is used to read the mail headers (plus the blank line separating the header from the body) and pipe them to the filter command.

Note that, if the filter decides that the mail should be retrieved, the complete mail has to be downloaded, including the headers, so the headers will be downloaded twice. This is because there's no way in POP3 to download just the mail body. Sometimes this overhead surpasses the savings of the filtering.

The filter command looks at the mail headers and signals with its exit code what mpop should do with the mail:

- 0: retrieve the mail
- 1: delete the mail; do not retrieve it
- 2: skip the mail; do not retrieve it

Return codes greater than or equal to 3 mean that an error occurred. The `sysexits.h` error codes may be used to give information about the kind of the error, but this is optional.

Since the filter command will be passed to a shell, you can use all shell command constructs in addition to just calling a script or program. This allows flexible filter constructs. See Section 11.2 [Filtering with SpamAssassin], page 23.

Some POP3 servers count end-of-line characters as two bytes (CRLF) instead of one (LF), so that the size of a mail as reported by the POP3 server is slightly larger than the actual size. The filters use the size values reported by the POP3 server since they cannot know the actual size in advance. Thus you cannot rely on *exact* size filtering.

11 Examples

11.1 A configuration file

```
#
# Default values for all accounts.
#

defaults
# Activate TLS.
tls on
# Enable full TLS certificate checks.
tls_trust_file /etc/ssl/certs/ca-certificates.crt
# Use the POP3-over-TLS variant instead of the STARTTLS variant.
# This is also known as "POP3 with SSL". Most servers support this.
tls_starttls off
# Use the procmail mail delivery agent.
delivery mda "/usr/bin/procmail -f '%F' -d $USER"

# For Sendmail:
#delivery mda "/usr/sbin/sendmail -oi -oem -f '%F' -- $USER"
# For msmtp (delivery via SMTP):
#delivery mda "/usr/bin/msmtp --host=localhost --from='%F' -- $USER"
# Delivery to a maildir folder:
#delivery maildir ~/Mail/incoming
# Delivery to a MBOX mail folder:
#delivery mbox ~/Mail/new
# Delivery to an Exchange pickup directory:
#delivery exchange c:\exchange\pickup


#
# Two pop3 mailboxes at the provider.
#

account provider1
host mx.provider.example
user john_smith
password secret

# Copy the settings from the previous account, and only override the
# settings that differ.
account provider2 : provider1
user joey
password secret2
```

```
#
# A freemail service.
#

account freemail
host pop.freemail.example
user 1238476
passwordeval gpg -d ~/.mpop.password.gpg

# The service runs SpamAssassin, so test each mail for the "X-Spam-Status: Yes"
# header, and delete all mails with this header before downloading them.
filter   if [ "`grep "^X-Spam-Status: Yes"`" ]; then exit 1; else exit 0; fi


#
# Set a default account (optional).
#

account default : provider1
```

11.2 Filtering with SpamAssassin

Use the following to delete all mails that SpamAssassin classifies as spam:

```
filter "/path/to/spamc -c > /dev/null"
```

Since no message body is passed to SpamAssassin, you should disable all body-specific tests in the SpamAssassin configuration file; for example set use_bayes 0.

If your mail provider runs SpamAssassin for you, you just have to check for the result. The following script can do that when used as an mpop filter:

```
#!/bin/sh
if [ "`grep "^X-Spam-Status: Yes"`" ]; then
    exit 1  # kill this message
else
    exit 0  # proceed normally
fi
```

Since the filter command is passed to a shell, all shell constructs are usable, so you can also use this directly:

```
filter if [ "`grep "^X-Spam-Status: Yes"`" ]; then exit 1; else exit 0; fi
```