CleanMail Home Version 5 Manual

Byteplant GmbH

May 8, 2012

Contents

Chapter 1

Introduction

Spam wastes time, clogs mail servers, can slow your server to a crawl, and is very difficult to get rid of. Most mailboxes today are constantly flooded with SPAM - unwanted advertising of any kind. Today the majority of all emails worldwide are spam mails.

While there is no shortage of solutions to this ever-growing problem, installing, using, and working with them often proves to be very complex. CleanMail is the mail filter software that was designed from the beginning to make installation, configuration, and maintenance as simple as possible.

The **CleanMail** product family brings the power of the award-winning open-source spam filter SpamAssassin™[1] to the Windows®[2] environment.

The filter pipelining architecture makes CleanMail a flexible multi-purpose mail processing tool. It allows for an easy integration of additional filtering programs like virus filters into the SMTP/POP3 checking pipeline.

These filter types are included in **CleanMail**:

Blacklist/Whitelist Filter Blacklist and whitelists allow filtering based on the sender address of a message.

Delay Filter (SMTP filtering only) The delay filter has proven to be very effective against the bulk mailer software used by spammers.

DNSBL Filter The DNSBL filter (also known as remote blacklist filter) can get rid of spam messages at the cost of a few DNS lookups.

Attachment Filter The attachment filter can remove potentially malicious attachments at very little processing cost.

[1] SpamAssassin is a trademark of the Apache Software Foundation

[2] Windows is registered trademark of Microsoft Corporation

Virus Filter CleanMail uses ClamWin (Clam Anti-Virus) to protect you from email-borne viruses. It also supports many third-party virus scanners out-of-the-box, e.g. Computer Associates Anti Virus, F-Prot Anti Virus, Kaspersky, NOD32, just to name a few. Virus mails are rejected and deleted by default.

SpamAssassin Filter SpamAssassin is the world-leading open source spam filter. Though it is one of the best spam filters around, with a very good spam detection rate and only few false positives, it processes mails only slowly and causes a rather high CPU load.

Spamtrap Filter (SMTP filtering only) This filter can be used together with SpamAssassin to train the spam mail database (Bayes database) used by SpamAssassin.

Mail Storage Use this filter to store verbatim copies of incoming messages somewhere on your hard disk or on network attached storage, in a folder you can configure.

External Commandline Filter This is the swiss army knife of mail filtering. You can supply your own home-made filters, and integrate them easily into the filtering pipeline of CleanMail.

1.1 POP3 Filtering

If you use POP3 filtering, CleanMail acts as a transparent proxy for your Internet service provider's POP3 server. The connection is initiated by your mail client connecting to CleanMail, and CleanMail forwards this connection request.

All messages have already been accepted and acknowledged by your ISP's mail server, so CleanMail is unable to reject the messages received. For this reason there can be no feedback to the sender of a message, if a message is classifed as spam and deleted

CleanMail is designed to work with all known POP3 servers, and with all mail clients supporting the POP3 protocol. This includes popular mail clients as MS Outlook, Mozilla Thunderbird, Eudora, or The Bat!.

Note that the APOP and IMAP protocols are not supported. Mail retrieved using these protocols is not filtered by CleanMail.

1.2 SMTP Filtering

The best place to stop SPAM is at the mail server, for two reasons:

- Spam mails can be deleted outright, before they enter your system. This saves your money, as you need less storage, bandwidth, and less of your users' time.
- If a legitimate email is identified as spam (false positive), the sender can be notified that his message might not be read by the recipient.

SMTP filtering is supported by CleanMail Server only, and not available in CleanMail Home.

1.3 Overview

Installation procedures and recommended network configurations are covered in *Installation* (chapter 2). This chapter also introduces the CleanMail application. This application gives you access to all configuration options and lets you view the CleanMail filtering status and statistics.

To learn about the concepts implemented in CleanMail, see *Concepts* (chapter 3).

Configuration of CleanMail is described in *Configuration* (chapter 4).

To find out what happens to your mail, CleanMail offers a lot of useful monitoring and reporting features. Learn about these capabilities in *Using CleanMail* (chapter 5).

See the *Reference* (chapter 6) chapter for details about the structure and content of the CleanMail configuration file.

See *Licensing* (chapter 7) for ordering and license details.

There are additional resources available online. Take a look at the FAQ list if you are running into problems. You will also find some 'How to...' documents there.

You may also want to look in the CleanMail support forum.

CleanMail support can also be contacted by email to support@byteplant.com.

Chapter 2

Installation

The installation section covers system requirements, CleanMail installation, and CleanMail uninstallation procedures.

CleanMail installs itself as an autostart program (located in the auto start subdirectory of the start menu). CleanMail is usually minimized, displaying a small icon in the system tray (usually in the lower right corner of the desktop). By right-clicking this icon, you can pop up a menu to maximize CleanMail.

CleanMail runs as a Windows Service. Thus Windows 2000/XP/2003/Vista/2008 is required.

Note: Depending on its configuration, Windows XP sometimes hides this icon. To make the CleanMail icon reappear, click on the arrow symbol left of the system tray.

2.1 Quick Start Installation Procedure

This document explains how to configure CleanMail Home for use with a POP3 email client like

- Microsoft Outlook/Outlook Express
- Eudora
- Pegasus
- The Bat!
- and many others more.

2.1.1 CleanMail Installation

When you start CleanMail for the first time, the quick start wizard will guide you through the installation process.

- Download CleanMail
- Install CleanMail using the *setup program* (section 2.2)
- Launch the CleanMail Admin application.
- The Quick Start Wizard will appear. On the POP3 settings page, follow the instructions to configure your mail client.
- Step to the following pages, and make adjustments according to your wishes. Save the configuration by pressing the 'Finish' button on the last page.
- Now retrieve a couple of mails from your mail box, if necessary, send yourself a couple of test mails. If you run into troubles, see *Troubleshooting the Installation* (section 2.3) for troubleshooting tips.
- In your mail client, add mail filtering rules to automatically move spam mails to a separate mail folder, or to automatically delete them. Refer to the manual of your mail client for instructions on how to do this.

Figure 2.1: Mail Path with CleanMail Filtering

2.1.2 Configuring Your Email Client for CleanMail

When the quickstart wizard is finished, you have to change the mail account settings in your email client software. Go to the mail account settings of your email client. Usually you will find the following settings:

- Outgoing mail server (SMTP server): Do not modify this setting, CleanMail does not interfere with outgoing mail.
- Incoming mail server (POP3 server): Write down this setting, and modify it to `localhost` or `127.0.0.1`. Make sure you use the POP3 protocol to fetch mail.
- User (Account): Modify this setting to username:mailserver, using the mail server name you wrote down in the previous step.
- Password: Leave this unchanged.

Note that CleanMail does not support the IMAP protocol. If your mail client is configured to use IMAP, reconfigure it to use POP3.

Repeat this procedure for all mail accounts and mail clients you use. Test your new settings immediately: Send yourself a test mail. Use your account to send a message to yourself, or use an e-mail echo server (e.g. echo@tu-berlin.de).

If you want to uninstall CleanMail later, remember that CleanMail's uninstall program will not reset the POP3 port of your mail client back to 110, you have to do this by hand.

2.2 CleanMail Setup

CleanMail setup features a standard Microsoft Windows®setup interface and you need only complete a few steps. You can cancel setup at any time by clicking the 'Cancel' button.

Double click `cleanmail.exe` (or similar filename) file on either the distribution media or from the downloaded `.ZIP` file. This will launch the CleanMail Setup Wizard.

Click 'Next' on the Welcome screen.

Read the CleanMail license and click 'I accept' to agree with this license.

Choose a folder where CleanMail should be installed. The setup program will suggest a default location. If you do not want to use the default location, you can browse for a specific directory in the provided input field (placing CleanMail in a location other than the default will not affect the operation of the program). Unless your CleanMail directory already exists (either the suggested, default directory or one of your choosing), the setup program will ask you if it can create that directory. Click 'Yes'. If you want to change the location of the program, click 'No'. This will keep you on the directory screen to choose another location.

The next step is to decide upon the name of the CleanMail program group name that you will see in the Start Menu. CleanMail suggests a default, but you can

change that to whatever name you would like (changing the name of the CleanMail program group will not affect the program operation in any way). After you have decided upon a name, click 'Next'.

There are some optional CleanMail Setup tasks that you may choose to have done. You can select these tasks by clicking on the appropriate check-box:

- Install additional ruleset - Installs additional spam filtering rules, not part of the SpamAssassin distribution
- Create a desktop icon - put a shortcut for CleanMail Administration Wizard on your desktop
- Create a quick launch icon - put a CleanMail Administration Wizard icon into the quick launch bar
- Windows Firewall Setup - check this to create Windows Firewall exceptions that allow mail transfers to pass through CleanMail (not for all versions of Windows)

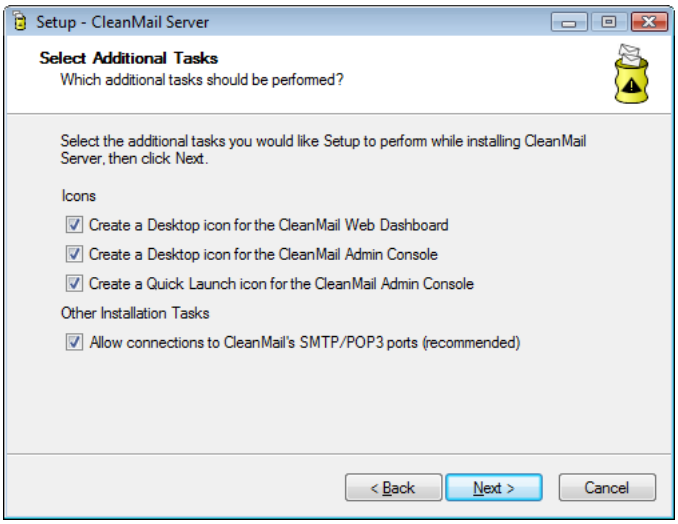

Figure 2.2: Installation Options

Click on the 'Next' button to continue. CleanMail will now install the program files and options. If there were no problems during installation, you will see the Finish screen. From here you can launch the CleanMail Administration Application. If you don't want to launch CleanMail, un-check the corresponding checkbox. Alternatively, you can just start the CleanMail service. Click the 'Finish' button when done.

2.3 Troubleshooting the CleanMail Installation

This section is intended to help you if you run into trouble during installation. If this section does not help with your problem, please consult the FAQ list. This page also offers access to some 'How to..' documents.

2.3.1 About Sockets, Ports, and Listeners

A network server is constantly ready to receive incoming connections from network clients. In other words, it is *listening*.

An SMTP server will be listening on the SMTP port of your machine, waiting for incoming connections from other mail transfer agents (MTAs, mail clients, or other mail servers) to send mail. SMTP (simple mail transfer protocol) is used to forward or deliver mail, and cannot be used to fetch mail.

POP3 servers listen on the POP3 port, providing a service for mail clients only. Mail clients can use POP3 (post office protocol version 3) to lookup if there are new mails available, and to fetch mails. POP3 cannot be used to send mail.

Ports are identified by port numbers: by convention the SMTP listening port is port 25, and the POP3 listening port is 110.

As a rule, only one program can be listening at the same time at any given port and IP address combination.

If you run CleanMail, and a mail virus checker on the same machine, this can be the first trouble you may run into: Both the mail virus checker and CleanMail contend for the POP3 port of your machine, but only one can use it, while the other fails to initialize. If CleanMail fails to grab the port, it will write an error message in its log file (look for a 'address in use' message in `cleanmail.log`) and pop up a warning dialog.

To fix this, set the POP3 port setting of your mail client to 111, and set the POP3 port number of CleanMail to 111, too (the setting can be found on the last page of the configuration wizard). This way you will get a checking pipeline like this:

mail client - (port 111) - CleanMail - (port 110) - antivirus - (port 110) - ISP POP3 server

If you want to uninstall CleanMail later, remember that CleanMail's uninstall program will not reset the POP3 port of your mail client back to 110, you have to do this by hand.

2.3.2 Testing the Basic Proxy Setup

POP3 communication was designed to be readable by human eyes. Because of this, the ubiquitous telnet program proves most useful to test your setup. In Windows,

you can run telnet from the start menu (choose 'Execute', and type 'telnet'), or from the command prompt. Try to connect with telnet to both your ISP's POP3 server and to CleanMail. Once you have seen the POP3 server's welcome message (starting with `+OK`), issue a `QUIT` command. Here is the transcript of a sample telnet session:

```
C:\>telnet localhost 110
Trying 127.0.0.1...
Connected to test1
Escape character is '^]'.
+OK CleanMail POP3 proxy ready.
QUIT
+OK closing connection
Connection closed by foreign host.
```

If everything works, you will get exactly the same replies both times. If you can't connect to the ISP's POP3 server, troubleshoot your firewall configuration, or it might be the POP3 server or the connection to the Internet is down. If you can't connect to CleanMail, look into CleanMail's log file for an error message.

2.3.3 Firewall Configuration

Once you have installed CleanMail Home, it will be CleanMail that connects out to your Internet service provider's POP3 server, and no longer your mail client. For CleanMail to work properly, you need to allow this connection in your firewall configuration. To find out how to do this, consult the documentation of your firewall software.

If CleanMail is unable to connect to the POP3 server, you will find 'unable to connect to outgoing server' messages in `cleanmail.log`.

2.4 Registering CleanMail

To register CleanMail, enter the registration name and license key you received when you purchased in the registration window. To make sure you enter the license key correctly, use copy/paste (CTRL-C and CTRL-V keys).

To obtain a license key, please visit our online shop.

2.5 Uninstalling CleanMail

When uninstalling CleanMail, do not forget to undo any changes you might have made in your firewall configuration or mail client configuration. CleanMail itself

can be uninstalled in one of two ways.

2.5.1 Using CleanMail Uninstall

This program is located in the CleanMail program group (the program group name may be different if you chose another name during setup). You can access it through the Start menu: Find and select 'Uninstall CleanMail' to run the uninstall program.

You will be asked if you want to completely remove CleanMail and all of its components. Click 'Yes' to continue with the de-installation or 'No' to cancel. If you click 'Yes', all installed files will be removed, any configuration files you created will be preserved. If removal was successful, a success message will appear (if you encounter problems during de-installation, please visit the Trouble Shooting section of this manual). Click okay to close this message. CleanMail is no longer installed on your computer.

2.5.2 Using The Microsoft Windows Control Panel

Select 'Add or Remove Programs' icon and then CleanMail. This will launch the CleanMail uninstall program. Follow the process as described in the *previous section* (section2.5.1).

Chapter 3

Concepts

This chapter is intended to help you understand the basic concepts in the design of CleanMail.

3.1 CleanMail Architecture

The CleanMail email security package consists of several parts:

CleanMail Service The main part of the package is the CleanMail Service. It runs in the background, intercepting mail transfers, and dispatches mail filters as needed. It also offers HTTP access to reporting pages you can access with a web browser.

CleanMail Admin The admin application allows to edit the configuration files used by the service. It can be used to define connectivity settings and filtering rules, and also displays runtime information such as statistics, system load, or the CleanMail Service's logs. If you have a mail storage configured, it also lets you browse stored messages.

SpamAssassin This is a filtering package from the SpamAssassin open source project. CleanMail supports two configurations: With Client/Server filtering, CleanMail will run one or more spamd filtering daemons, and uses the spamc program to submit mails to spamd for filtering. If Client/Server filtering is disabled (the default) the CleanMail service will run a `SpamAssassin` executable for every message to filter.

ClamWin This an open source virus scanner package you can choose to install with CleanMail. It is invoked whenever needed from the CleanMail service to filter messages. The ClamWin package contains its own maintenance and support programs to update the filter database, or to check your hard disks for virus infections.

3.2 Proxy Ports

A proxy is a server that sits between a client and a server. The proxy intercepts all requests to the server to either handle them by itself or to forward them to the server.

With POP3 filtering, the client is your mail client, trying to fetch mail, and the server is your ISP's POP3 server.

The mail client that connects to CleanMail does not see any difference in the service your ISP's POP3 server usually provides.

3.3 Filter Pipeline

CleanMail feeds incoming mail to a series of mail filters, the so-called filter pipeline. Examples of mail filters are the built-in attachment blocker, third-party virus checkers, or SpamAssassin.

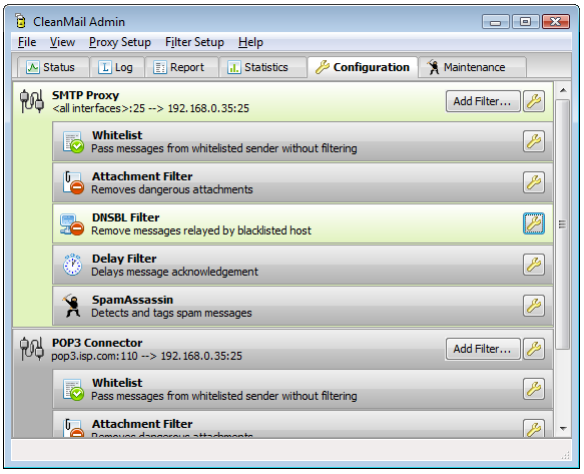

Figure 3.1: Example Filter Pipeline

Each filter analyzes the message and returns a filter result telling CleanMail what to do with it. Example filter results are: accept and deliver, or reject and delete. The overall filtering result is always the "worst" result, for example if the virus checker returns 'reject and delete', it will override another filter returning 'accept and deliver'.

The location of a filter in the filter pipeline matters: To help conserve resources and to increase throughput, filters lower down in the filter pipeline are not invoked if an earlier filter has already decided that a mail should be deleted. Filters are also skipped if the filtering result so far can't be exceeded by the worst result a filter may return.

CleanMail by default orders the filters to optimize throughput, using the following guidelines:

- Filters with the lowest resource usage and the highest selectiveness should go first. For this reason the attachment filter is always be one of the first filters in the filter pipeline, because of its low resource usage and its good results in finding and removing virus mails.
- Filters which use a lot of processing power and with low selectiveness should go last. Most people won't delete spam mails without at least giving humans the chance to look them over: For this reason, the selectiveness of the SpamAssassin filter is low, while it uses a lot of resources. Therefore, the SpamAssassin filter should be one of the last filters.

When configuring CleanMail with the Admin application, every new filter will be automatically moved to the best position in the filter pipeline. Afterwards, you can still change the order of filters, but only within limits. You can also move filters as you please (though we do not recommend it), by editing the configuration file with a text editor (see *CleanMail Configuration File* (section 6.1)).

3.4 Mail Storage

You can configure a mail storage in a proxy port's filter pipeline. The mail storage takes a copy of the message it processes and saves it on disk. You can later browse stored messages, and view messages or message transport information.

Like with filters, order matters: If you choose to place the mail storage before the first filter of you pipeline, all messages, including each and every spam message are saved. Put a mail storage here if you want to be able to retrieve a copy of false positives.

If you place the mail storage last, only messages actually delivered are stored. You can use this to archive messages.

3.5 Remote Access

The CleanMail Service allow remote access using the HTTP protocol to view logs and statistics. Access is read-only in general, with some enhanced functionality to browse the mail storage (if configured). Administrators may choose to restrict access by using passwords, and by using host allow/deny lists.

To avoid port conflicts, CleanMail Service listens by default on a non-standard port (8086) for HTTP connections.

Chapter 4

Configuration

CleanMail is usually minimized, displaying a small icon in the system tray (usually in the lower right corner of the desktop). By right-clicking this icon, you can pop up a menu to maximize CleanMail, giving you access to the configuration dialogs.

Note: Depending on its configuration, Windows XP sometimes hides this icon. To make the CleanMail icon reappear, click on the arrow symbol left of the system tray.

Within the configuration dialogs, you can freely step forward and backward using the 'Next' and 'Back' buttons. You can cancel your changes anytime by pressing 'Cancel'. Once you pressed 'Finish' your changes become permanent and are stored in CleanMail's configuration file, `cleanmail.cf`.

Important: Once you have saved new settings, they are not yet in use by the CleanMail service. To make the service re-read the configuration file, choose 'Apply Settings' from the file menu once you are ready.

4.1 POP3 Proxy Port Setup

The POP3 Proxy Port Setup dialog is invoked by choosing "Global Settings" from the file menu, or by double-clicking the POP3 proxy port information on the configuration page.

4.1.1 POP3 Server and Port Settings

This page allows to set the basic connectivity settings of the proxy port you are configuring.

Incoming IP Address/Port

Choose one of the IP addresses available. Use '`<all interfaces>`' if you want the proxy to listen on all interfaces. This setting will make CleanMail listen on all IP addresses, including the loopback interface (127.0.0.1).

Usually, the port number will be the POP3 port number, 110.

Outgoing IP Address/Port

The outgoing IP address cannot be configured in advance. It depends on the account a user wants to connect to, and it is specified in the mail client (see below). The outgoing port number is always set to 110.

4.1.2 Changing the Mail Account Settings

If you want to use the POP3 proxy to filter incoming mail, you have to change the mail account settings in the configuration of the mail client software you use. Usually you will find the following settings:

Outgoing mail server (SMTP server): Do not modify this setting, the POP3 filter of CleanMail does not interfere with outgoing mail.

Incoming mail server (POP3 server): Write down this setting, and modify it to the hostname or to the IP address of your CleanMail server. Make sure you use the POP3 protocol to fetch mail.

User (Account): Modify this setting to *username*:*mailserver*, using the mail server name you wrote down in the previous step.

Password: Leave this unchanged.

Note that CleanMail does not support the IMAP protocol. If your mail client is configured to use IMAP, reconfigure it to use POP3.

Repeat this procedure for all mail accounts and mail clients you use. Test your new settings immediately: Send yourself a test mail. Use your account to send a message to yourself, or use an e-mail echo server (e.g. echo@tu-berlin.de).

4.1.3 Logging Options

The log output can be seen on the "Log" page of CleanMail Admin, or by viewing the file `cleanmail.log`.

The log file is cycled whenever its size exceeds the limit, or at midnight when a configurable number of days has passed. The verbosity of the log file is controlled by the following flags:

- Extended logging adds some more output to the log that might be interesting. Turning this option on will log the To, From, and Subject mail headers of every mail received.
- Detailed logging, among other things, adds a transcript of the entire POP3 communication to the log. This is most useful for debugging mail transport problems.
- Filter error logging collects error or debug output of the mail filters and writes it to the log file. Virus filters often log the type of virus found to their error output. With this option on, you can see it in CleanMail's log.

The **Hide Receive Mail Status** setting disables the small popup window that appears in the lower right corner of the desktop, displaying the mail download progress.

4.2 Mail Filter Setup

This section discusses filter settings common to all filters. Please read *Filter Pipeline* (section 3.3) for an intorduction to filter pipelines.

4.2.1 Filter Name

Filter names are used to identify individual filters in statistics charts and reports. The name should be unique.

If you do not have more than one filter of the same type, there is usually no need to override the default name. Therefore, this setting is usually hidden. It is only accessible once you have more than one filter of the same type in use.

4.2.2 Filter Results

All filter configurations have a setting that allows you to choose what happens with a mail if the filter finds unwanted content, such as a virus, or spam. The following summarizes the filter results you may encounter, and their reasons:

accept/deliver The filter did not find unwanted content.

accept/deliver (skip size exceeded) Some filters do not check mails larger than a configurable size. For example, spam mails are typically small, so the SpamAssassin filter by default passes large mails without checking.

accept/deliver (junk) The filter found unwanted content, but the mail is accepted and delivered nonetheless. If the **Subject Tag** setting is not empty, CleanMail will flag the message as junk by modifying the subject.

accept/deliver (unknown result) For some reason, the filter was unable to check the message. The filter will write additional information about the problem to `cleanmail.log`. The mail is accepted and delivered.

accept/delete The filter found unwanted content. Receipt of the mail is acknowledged, but the mail is deleted. The mail simply vanishes, the sender is not notified, and the recipient never sees it.

accept/deliver (whitelisted) The sender address is whitelisted. All filters (except attachment blockers and anti virus filters) are bypassed, and the mail is accepted and delivered.

accept/deliver (license count exceeded) The filter was disabled because the recipient address count covered by your license was exceeded. The mail is accepted and delivered.

All messages that have been processed by CleanMail will have a "X-CleanMail-Result" header field. This can be used by the mail client or server to quarantine or delete mails. See your mail software's documentation to find out how to set up filtering rules.

Note: Filter results lower down in the list take precedence. Filters further down in the filter pipeline can override results of earlier filters. For a discussion of this, see *Filter Pipeline* (section 3.3).

4.2.3 Subject Tag

Set the text added to the subject of mails considered spam. This setting allows `US-ASCII` non-control characters only (character codes 32-127).

4.3 Attachment Filter Setup

Most worms and viruses are spread by attachments. Getting rid of messages containing potentially malicious attachments before even starting virus checkers or spam filters can be very helpful to reduce system load.

4.3.1 Attachment Filtering Options

CleanMail's built-in attachment blocker allows you to specify a list of attachments that you want to accept, or, the other way round - you can specify a list of attachments you do not want to accept. The attachment filter is pre-configured to reject all attachment types known as potential virus vectors.

The configuration page allows you to enter attachment types in three input fields. Settings in fields lower down override settings in the fields higher up. You can use the wildcard characters `?` (any character) and `*` (any number of any character), for example like in `vb*`.

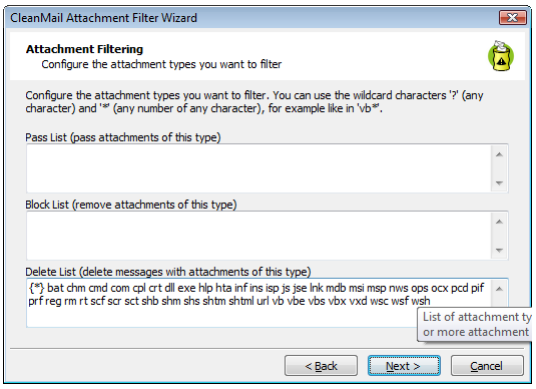

Figure 4.1: Attachment Filter Setup

Here is an example:

```
Pass Attachment List:   zip bmp
Block Attachment List:  *
Delete Attachment List: scr pif
```

This passes only zip and bmp attachments, all else are blocked. If a mail has a `pif` or `scr` attachment, the attachment is not only blocked, but the entire mail is deleted (`pif` and `scr` are very common as virus vectors).

4.3.2 Ignore Whitelist

For security reasons, anti virus filters and the attachment blocker by default ignore whitelisting. You can change this behaviour by removing the check mark from the **Ignore Whitelist** setting.

4.3.3 MIME Error Policy

MIME violations can disrupt mail server operation and sometimes crash mail clients. Also, worm authors try to hide executable attachments with deliberate MIME syntax violations.

The attachment blocker is also capable of detecting MIME violations, and you can choose which policy to apply for messages affected:

- **General MIME syntax violation (SEVERE)** Worm authors could try to hide executable attachments with deliberate MIME syntax violations.
- **ASCII-0 character (SEVERE)** The mail contains an ASCII-0 character. This problem can disrupt mail server operation and sometimes crash mail clients.
- **Line Break Error** The message has a single carriage return or line feed character in the message or a line is too long. These MIME violations are very common in spam messages, but also sometimes present in legitimate messages.
- **8-bit character in header** This MIME violation is very common and, for this reason, only reported in the log.

For severe MIME violations, we recommend deleting messages. You can choose a different setting, if desired. The policy you apply has to be chosen with the **MIME Error Policy** setting.

The **Line Break Policy** setting controls the handling of wrong line breaks. Messages with wrong line breaks are passed by default.

4.4 Blacklist Filter Setup

The Blacklist filter uses static address patterns to check the sender address fields of a message. A blacklist filter is very time-consuming to maintain, and usually not very effective, as spammers can easily fake a different sender address.

4.4.1 Sender Address Patterns

This is the list of sender addresses or address patterns to blacklist. You can use the wildcard characters '?' (any character) and '*' (any number of any character), for example like in `*@obnoxious.site`. Addresses you enter here are automatically normalized (lower-case characters), and sorted by domain.

4.4.2 Policy

The policy you choose in this setting is applied when a sender matches one of the sender address patterns.

4.5 Whitelist Filter Setup

The Whitelist filter uses static address patterns to check the sender address fields of a message. Mail from whitelisted addresses is accepted and bypasses all filters but the attachment filter and anti-virus filter. (You can configure those filters to pass whitelisted messages as well.)

4.5.1 Sender Address Patterns

This is the list of sender addresses or address patterns to whitelist. You can use the wildcard characters '?' (any character) and '*' (any number of any character), for example like in `*@byteplant.com`. Addresses you enter here are automatically normalized (lower-case characters), and sorted by domain.

Important: For security reasons, anti virus filters and the attachment blocker by default ignore whitelisting. You can change this behaviour by removing the check mark from the **Ignore Whitelist** setting in these filters.

4.6 RBL Filter Setup

DNS blacklists are Internet resources maintaining databases of known spam relay hosts. Mail servers can query these databases in an efficient manner using the DNS (domain name service) protocol. The RBL filter rejects all mail that has been relayed by a blacklisted host.

The RBL filter is highly efficient, and typically capable of getting rid of half the spam messages at a low resource usage.

4.6.1 DNSBL Zone List

This setting defines the DNSBL blacklists to query. If a relay host is listed in one of these zones, the message is blocked using the filter policy you can define below.

Choose with care, because picking the right zones affects the RBL filter's effectiveness. When in doubt, stick with the default setting.

4.6.2 Policy

If a relay host is listed on one of the configured DNS blacklists, the mail is blocked using this policy.

4.6.3 Relay Check Option

All relays forwarding a mail message prepend a new received header field to the mail header, with information about the servers involved (host name and IP), the protocols used, and a time stamp.

The header lines are parsed by the RBL filter to find the IP addresses to check against the DNS blacklists defined in the **Zone List** setting. The last received header (at the top of the message header) marks the transfer to the server handling final delivery, and the first received header contains information about when a message was first submitted for transmission to an SMTP server. Between these entries, there can be any number of relays forwarding the message. The first and last received headers can be the same, when a message was directly submitted to your server.

The **Relay Check Option** defines which received headers (relays) are checked by the filter. If DNS blacklists contain the IP addresses used by dial-up services, you can reduce the risk of false positives by skipping the DNSBL check for the first received header (created when a dial-up sender submits his message to the first SMTP server). Note that the last received header (final delivery) will always be checked, even when there is only one received header. Allowable values are `all`, `all but first`, and `last only`. The default setting is `all but first`.

4.7 Shared Real-Time Fingerprint

The fingerprint filter calculates message fingerprints and compares them against a database of known spam message fingerprints. If a message is blocked by another filter, its fingerprints are automatically added to the database, speeding up the processing of similar spam messages.

4.7.1 Fingerprint Database

The fingerprint filter maintains a file of known message fingerprints. This file is continuously updated using the results of local message processing (using your filter pipeline), and from a remote database, which is queried in regular intervals. The remote database is a central database maintained by Byteplant and available from our servers for use of all CleanMail customers.

Communication with the remote fingerprint database is two-way:

Shared Real-Time Fingerprint Filtering Any new spam fingerprints detected by local filtering are used to refine and improve the remote fingerprint database. Local filtering automatically improves filtering and performance for all CleanMail customers.

Firewalls and HTTP Proxies The fingerprint filter uses the HTTP protocol to access our web servers for the exchange of fingerprint data. Please allow the CleanMail process to access http://www.byteplant.com in your firewall configuration. If you use a HTTP proxy, be sure to configure the **Check For Updates Proxy** setting on the admin mail options settings page.

Privacy Fingerprint data submitted to our server does not allow to reconstruct the contents of a message, or to identify the senders or recipients of messages. Access to the fingerprint server is logged using our standard web server logging policy. Please see http://www.byteplant.com/company/privacy for details.

Abuse The fingerprint database server has safeguards in place to prevent abusive use or manipulation of the database.

4.7.2 Fingerprint Filter Settings

4.7.2.1 Policy

If a message fingerprint matches a known spam message fingerprint, the mail is blocked using this policy.

4.7.2.2 Skip Size

Spam and virus messages are usually small. To conserve system resources and increase throughput, it is recommended to skip filtering mails exceeding a certain size.

4.8 External Filter Setup

The external filter is used to pass a mail through another program. The SpamAssassin filter and the anti virus filter both are based on external filters.

External filters can be used for many tasks, such as archiving mails, providing additional statistics, or storing mails in an SQL database.

The CleanMail development team is ready to provide custom-made filters designed to fit your specifications.

4.8.1 Command Line

The command line setting page is the core of external filter setup. Here you can control which program to run with what arguments.

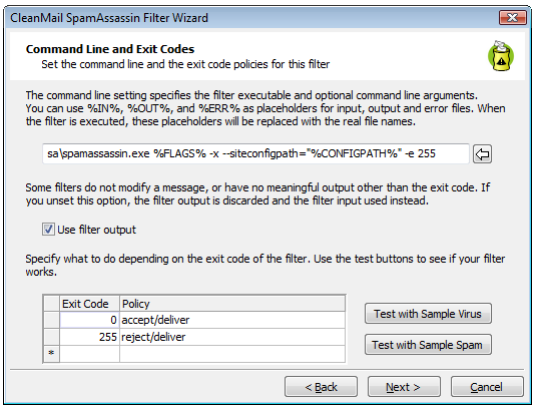

Figure 4.2: External Filter Setup

4.8.1.1 Message Text Input and Output

The message to check is passed to the standard input of a external program.

If a program does not modify a message (virus checkers, for example, only analyze a message), you can choose to ignore the output of a program, in this case the unmodified message will be forwarded to the next filter. Otherwise, standard output of the program will be used.

If you've enabled filter error logging (see *Global Settings* (section4.1.3)), the standard error output of a program is collected and printed to the log.

Instead of piping the message through standard input or output, you can use the placeholders `%IN%`, `%OUT%`, and `%ERR%` as arguments to a program. Here is an example:

```
"c:\dir\clamscan.exe" "%IN%" --mbox --no-summary
```

Notes:

Be sure to use double quotes where needed. If your temporary directory path, for example, contains blanks, `%IN%` must be quoted, because the filter input and output files reside in the temporary directory.

The working directory of external programs always is the CleanMail installation directory.

To test your settings with a sample spam mail or a sample virus mail, use the test buttons provided.

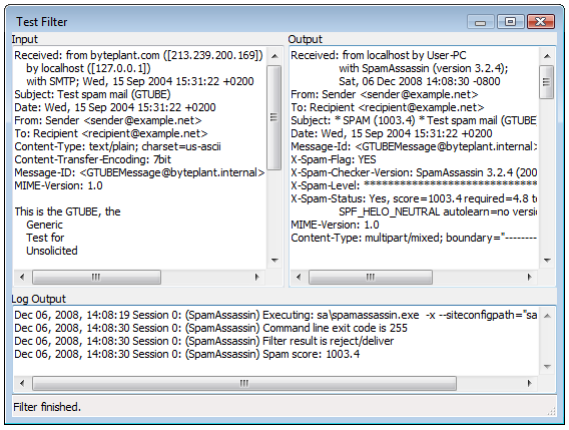

Figure 4.3: Test Filter Screen

4.8.1.2 Batch Files

You can also run batch files instead of running a program. Depending on the Windows version you are using, it may be necessary to explicitly run a command line interpreter with your batch file as an argument. Here is an example what this might look like for Windows XP:

```
cmd /Q /D /C "c:\dir\batch.bat" %IN% %OUT% %ERR%
```

To learn more about `cmd`, type `help cmd` in a command prompt window.

4.8.2 Advanced Settings

The advanced settings allow choosing a timeout for the external program, a size limit, and setting the memory usage.

4.8.2.1 Timeout

If the external program does not return a result within the set timeout period, the program is terminated and the filter result is set to *accept/deliver (unknown result)* (section4.2.2).

The program will also be terminated when the SMTP session times out or if the MTA that connects to CleanMail disconnects. The SMTP timeouts used by most MTAs are in the range of 5 to 10 minutes.

If you set a timeout, a value in the range of 3-4 times the normal execution time is advisable. External filter programs should not take longer than 20 seconds to execute.

4.8.2.2 Ignore Whitelist

Use this setting to configure if the filter should run for whitelisted senders or not. This setting is disabled by default for all types of external filters, with the exception of anti-virus filters, where it is enabled by default for security reasons.

4.8.2.3 Skip Size

Spam and virus messages are usually small. To conserve system resources and increase throughput, it is recommended to skip filtering mails exceeding a certain size.

4.8.2.4 Memory Usage

Specify here how much system RAM your filtering program needs. This setting helps CleanMail to optimize resource allocation. Worst-case memory usage of virus checkers for large mails is usually about 3-4 times the size of the checked mail.

4.8.3 Return Code Policy

On this page, configure the mapping of exit codes (0-255) to filter results.

Clam Anti Virus, for example, returns 0 if a mail is not infected with a virus, 1 if a virus is found. Therefore, the return code 0 is mapped to the result accept/deliver, and return code 1 is mapped to reject/delete.

Consult the sections on *filter pipelines* (section 3.3), and on *filter results* (section4.2.2) for more info.

4.9 Anti Virus Filter Setup

Every virus checker that offers a command line interface can be integrated into CleanMail. In the Anti Virus Filter Setup dialog you can easily adapt and test the operation of third-party virus checkers.

The Anti Virus Filter Setup dialog is based on the *External Filter Setup* (section 4.8) dialog. See there for an explanation of settings not explained here.

To configure a virus filter, CleanMail needs to know the vendor name. You can set the scanner path in the 'Scanner Executable' setting if you did not install your virus checker in its default location.

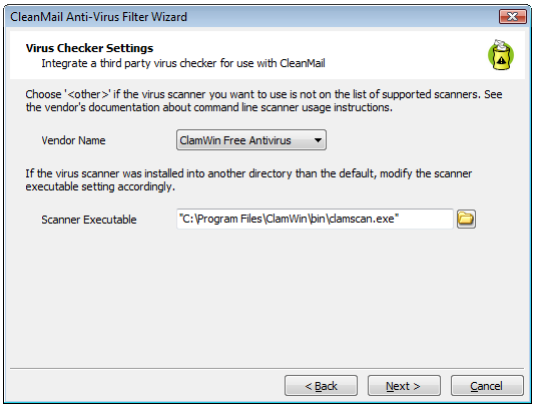

Figure 4.4: Anti Virus Filter Setup

If your virus scanner is not on the list of supported scanners, choose 'Other' on the first page of the setup dialog. Consult the virus scanner's documentation to find out command line options and usage instructions.

For most filters, the output of a filter is forwarded to the next filter as its input. This can't be applied to most virus checkers, because virus checkers analyze a message only, signalling "yes, this is a virus", or "no, this is not a virus" with different program exit codes. For this reason virus filters are by default configured in a way that the input file is forwarded to the output, by leaving the "use console output" switch unchecked.

Normally, you will want to delete a message if the return code of the filter indicates a virus has been found, and deliver a message if not. Set the return code policies accordingly.

When testing a virus filter, test it both against the sample virus mail, and against the sample spam mail. The virus mail must be blocked (reject/delete), with the filter output empty, while the spam mail must pass (filter output the same as the input).

Important: For security reasons, anti virus filters and the attachment blocker by default ignore whitelisting. You can change this behaviour by removing the check mark from the **Ignore Whitelist** setting.

Note 1: Integrating a virus checker in CleanMail requires that you install the virus checker software first. If you have not installed the virus checker yet, run the anti virus setup wizard again once you have installed the virus checker.

Note 2: If you want to use F-Prot for DOS, right-click `F-PROT.EXE` in the Windows Explorer to bring up the properties dialog, and make sure the 'close window on terminate' (or similar) property is checked.

Note 3: If you have a virus scanner with On-Access scanning enabled, it may interfere with the temporary files CleanMail creates for filtering in the temporary directory. To get rid of the error messages that may occur in CleanMail's log file,

make sure you disable On-Access scanning for the temporary directory used by CleanMail. On startup, CleanMail writes the location of the temporary directory used to the log file `cleanmail.log`.

Note 4: If you enabled POP3 mail virus checking in your anti virus software, don't integrate the virus scanner in the POP3 filter pipeline. Otherwise you might end up checking your mails twice. Note also, that in this case both CleanMail and your virus scanner may contend for the POP3 port of your machine. Read *Troubleshooting* (section 2.3) for more information.

4.10 SpamAssassin Filter Setup

The SpamAssassin filter setup dialog is available in two modes: normal and advanced. Advanced mode is based on the *External Filter Setup* (section 4.8) dialog. This section discusses normal mode setup.

The SpamAssassin filter setup dialog allows you to configure two aspects of SpamAssassin filters:

- A plugin part that controls to what messages the filter is applied and what is done with a mail, once it is tagged as spam (*filter policy* (section4.2.2)). The settings in this part are stored in the `cleanmail.cf` file.
- Configuration of SpamAssassin itself. The settings in this part are stored in the file `local.cf` in the SpamAssassin rule set directory.

It is important to remember this, especially when you are planning to use *multiple SpamAssassin filters* (section4.10.3.2) in your configuration.

4.10.1 How SpamAssassin Works

SpamAssassin is a well-known open-source spam detection engine. It uses the following techniques to identify spam:

- The mail headers are scanned for some small inconsistencies that can give away forgeries: A mail date in the past or in the future, forged message IDs, and the like.
- The mail body is scanned for typical spam mail content, such as spam keywords, capitalized letters, or invitations to buy or click something.
- Queries to blacklist servers are used, e.g. to see if a mail has been submitted from a known open mail relay.

- Probability analysis of mails (Bayes filtering). Spam mails can be trained, so that similar mails are more likely to be identified as spam in the future.
- Analysis of the URLs a mail refers to: Spammers want you to click on hyperlinks referring to their sites, so a lookup in a database of known spam-advertised sites has proven to be highly effective in identifying spam mails.

The result of all these tests is added up to form a spam score. A message is considered spam if the score exceeds a configurable threshold. You can modify the aggressiveness of the spam checker by modifying this threshold: an aggressive setting with a low threshold will find more spam mails, at a higher risk of legitimate mails falsely identified as spam (false positives).

In the typical configuration, the subject of mail identified as spam is modified to flag it as spam mail and the message is quarantined within a SpamAssassin wrapper to prevent accidental infection with dialers, spyware, trojans, or viruses by viewing HTML content.

Now it is up to your mail server (or client) software to decide what to do. You can delete spam mails or move spam mails to a spam folder or leave the decision what to do with spam mails to your users.

4.10.2 SpamAssasssin Options

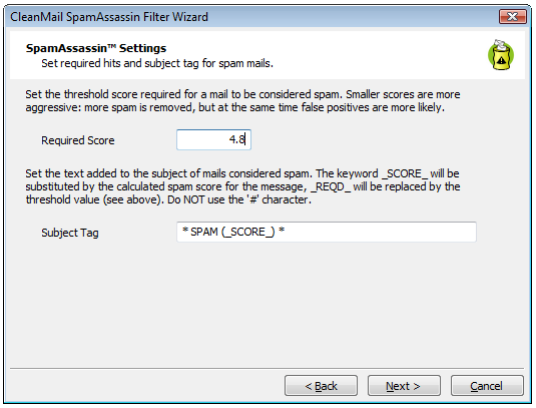

Figure 4.5: SpamAssassin Options Setup

4.10.2.1 Required Score

SpamAssassin tests each incoming mail against its spam detection ruleset. Each matching rule adds a predefined score to the overall spam score.

Set the threshold score required for a mail to be considered spam. The default setting is 5.0, which is quite aggressive, increase this value to reduce the probability of false positives.

4.10.2.2 Subject Tag

Set the text added to the subject of mails considered spam. The keyword `_SCORE_` will be substituted by the calculated spam score for the message, `_REQD_` will be replaced by the threshold value (see above). This setting allows `US-ASCII` non-control characters only (character codes 32-127). Do NOT use the # character.

Note: The SpamAssassin Filter is the only filter that supports substitutions in the subject tag (e.g. spam score).

4.10.2.3 Tweaking The SpamAssassin Rule Set

If you want to further customize SpamAssassin, consult the SpamAssassin documentation files included with the installation files (find it in the `sa\doc` subdirectory of the installation directory).

To customize the SpamAssassin rule set, for example to modify the score for a particular rule, you can do so by editing the corresponding configuration file using your favorite text editor. See the document `Mail_SpamAssassin_Conf.htm` for details.

Note 1: Configuration changes in files other than `local.cf` are not backed up upon installation of an update. If you want to keep your changes, copy the files you changed, and restore them after installation.

Note 2: To validate your changes, use the `--lint` option of SpamAssassin:

```
cd [InstallationDirectory]
sa\spamassassin -x --siteconfigpath="sa\ruleset" --lint
```

4.10.3 CleanMail SpamAssasssin Options

4.10.3.1 Spam Mail Policy Options

On the first page of the spam mail policy options you can choose what happens with spam. See *Filter Result* (section4.2.2) for a list of different spam mail policies.

The second page allows to set a Reject and Delete threshold. Mails are deleted (reject/delete policy) if the spam score is higher than this value, regardless of the policy setting you entered on the previous page.

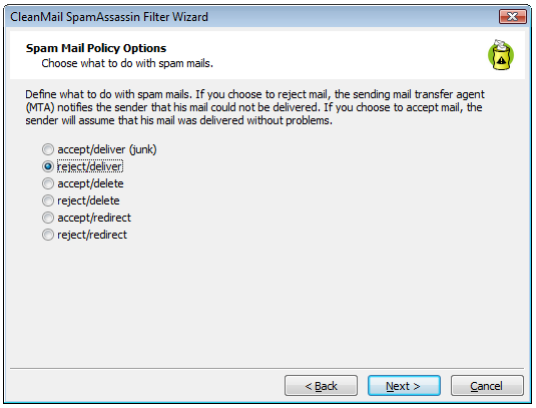

Figure 4.6: Spam Filter Policy Setup

4.10.3.2 Multiple SpamAssassin Filters

If you are using multiple SpamAssassin filters, by default all filters use the same SpamAssassin configuration settings, stored in the `local.cf` file of your default rule set directory (the `sa\ruleset` subdirectory of your installation).

If you intend to use different SpamAssassin configurations for your SpamAssassin filters, copy the `sa\ruleset` directory to a different directory and update the filter settings of the SpamAssassin filter that uses this directory accordingly (a special setup page to enter the directory becomes visible in the SpamAssassin filter setup wizard as soon as you add a second SpamAssassin filter to your configuration).

Note that CleanMail's installer only updates the default SpamAssassin rule set, never any additional rule sets. You may need to update additional rule set directories (add/remove/update configuration files other than `local.cf`) manually.

4.11 Mail Storage Setup

The mail storage filter can be used to archive mails on the file system of your server. The mail files (`*.eml`) are stored in MIME-Format (RFC-822) and can be viewed with the majority of mail client software. In addition to the mail file, message transmission data, such as the SMTP sender and recipients, is saved in an `.envelope` file. The envelope file can be viewed with a text editor.

Many of the message browsing and learning functions (see *Learning Messages* (section 5.6)) require that a message has been saved with a mail storage filter.

4.11.1 Storage Directory

Sets the directory where the mail files will be stored. If empty, the system temporary directory is used.

Note 1: Make sure that CleanMail (or the account used by the service, normally the system account) has access permissions to create, read, write, or delete files in the target directory. This is especially important if the target directory resides on a network drive.

Note 2: Make sure that unprivileged users do not have access to this directory, otherwise a user's mail would be readable by others.

Note 3: If you have a virus scanner with On-Access scanning enabled, it may interfere with the mail storage whenever a virus message is stored. As a result you may see error messages in CleanMail's log file.

4.11.2 Max. No. of Days

This sets the maximum number of days messages are kept in storage. Leave this setting empty if you don't want to use this feature.

4.11.3 Max. No. of Messages

This sets the number of messages to store in the cache. Once the limit is exceeded, the storage manager starts deleting old messages. Leave this setting empty if you don't want to use this feature.

4.11.4 Max. Cache Size

This sets the maximum disk space used by the cache. Once the limit is exceeded, the storage manager starts deleting old messages. Leave this setting empty if you don't want to use this feature.

Chapter 5

Using CleanMail

CleanMail offers a comprehensive suite of reports and statistics to allow monitoring its activities. You can find and view mail messages, whitelist mail sender addresses, and improve mail filtering by adding spam messages to its database of known spam message fingerprints, or to the SpamAssassin Bayes database.

5.1 Live CleanMail Status

The live status page is displayed when you launch the CleanMail Admin application. This page shows the network traffic of the CleanMail Service running in the background.

You can watch the traffic increasing whenever your mail client software fetches new mail from your ISP's POP3 server.

5.2 Server Log

To view the server log, switch to the **Log** tab of the CleanMail Admin application. This page is a live view of messages written to CleanMail's log file `cleanmail.log`, located in the installation directory.

The verbosity of the messages in the log can be modified using in the *configuration wizard* (section 4.1).

To view the logfile with your default text editor, press the 'View Log File With Editor' button.

To get a deeper understanding of POP3 and what happens in the log (especially when you enabled detailed logging), please refer to RFC-1939.

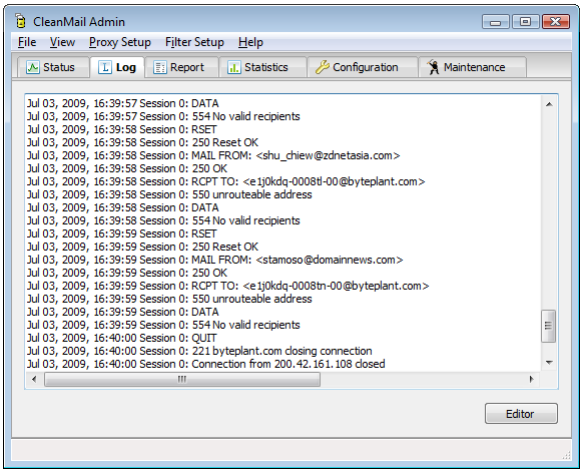

Figure 5.1: Log View

5.3 Report

A choice of reports can be viewed on the **report** tab of the CleanMail admin application.

Journal All messages handled in the past few days. The length of the backlog is limited and depends on mail traffic.

Top Recipients A list of people that received mail in the last 24 hours or yesterday, complete with message counts and the total number of bytes received.

Top Senders A list of senders that mailed to accounts in your domain in the last 24 hours or yesterday, complete with message counts and total bytes.

Top Hosts A list of mail hosts (IP address and name) that delivered mail in the last 24 hours or yesterday, complete with message counts and total bytes.

Top Spam Hosts A list of mail hosts (IP address and name) that delivered a spam message in the last 24 hours or yesterday, complete with message counts and total bytes.

For all reports, you can change the sort order by clicking the column headers. To filter the results, choose another time range, or a specific port, or enter a filter string.

The **Source** input field gives you a choice of data source. You can choose between the following:

Mail Log Displayed data is taken from CleanMail's *mail log file* (section 6.2). This includes all messages received or rejected in the last few days.

Mail Storage Displayed data is taken from the index of a mail storage. All messages listed are found in the mail storage. The time range of messages found here depends on the settings of the mail storage.

Search Displayed data is taken only messages that satisfy the search criteria you have defined and stored.

After selecting a line in a report, you can right-click the line or press the **Message** button to choose from a menu of actions:

Open Message The message is opened using the application associated with `.eml`. With many versions of Windows, the default application associated with this file type is Outlook Express. This feature requires that a copy of the message is cached in a mail storage. This action is **unsafe** to use with virus messages, depending on the associated application.

Unblock This is similar to **Send Message**, but the message is automatically forwarded to its original recipients, if this information is available. This feature requires that a copy of the message is cached in a mail storage. This action is **unsafe** to use with virus messages, depending on the mail client used by its recipients.

Send Message A verbatim copy of the message is submitted to a recipient of your choice, using a mail server of your choice. This feature requires that a copy of the message is cached in a mail storage. This action is **unsafe** to use with virus messages, depending on the mail client used by the recipient.

View Message The contents of the message are displayed in an internal viewer window. The display is plain ASCII text, no attachments are unpacked and no HTML code is interpreted, so it is safe to use this action with virus messages. This feature requires that a copy of the message is cached in a mail storage.

View Envelope Displays mail transmission data, depending on the transmission protocol used. This feature requires that a copy of the message is cached in a mail storage.

Transmission Log Searches the CleanMail log file to find the session that actually handled the selected message, and displays all information about this session. The transmission log may be unavailable if the log file has already been cycled. To get the most from this feature, enable detailed logging.

Copy To Clipboard Use this menu item to copy ie. the message subject, sender, or the recipient to the clipboard.

Learn Use these menu items to *train CleanMails spam database* (section 5.6), improving results for the fingerprint filter, and for the SpamAssassin filter. Learning a message requires that a copy of the message is cached in a mail storage.

Export Table... Export the data displayed to a file of comma-separated values (verb.csv). The exported file can be loaded into spreadsheet software such as Microsoft Excel for further processing.

Blacklist Messages using a blacklisted sender address are always blocked in the future. This rarely works, as spammers usually use a new sender address for every mail they send. If you use blacklist or whitelist filters, be sure to check and optimize these filters regularly. Bloated blacklists/whitelists may degrade overall performance. You can blacklist a sender even when the message is not stored.

Whitelist Messages sent from a whitelisted address are always passed in the future. Some filters such as Antivirus filters may choose to ignore this setting (you can configure this filter behavior). You can whitelist a sender even when the message is not stored.

5.4 Search

The **Search** page allows to save search configurations you can later access again, both from the report page and the web dashboard.

5.5 Statistics

CleanMail maintains a number of counters to collect statistics data, such as raw SMTP network traffic, mail counters, and filter result counters.

The **Statistics** of the CleanMail Admin application offers a graphical visualization of these counters: total mail received, mail passed, mail blocked. Historical data displayed is read from the *statistics file* (section 6.2). Additionally, the graph is continuously updated with live data.

5.6 Learning Messages

Sometimes spam messages are not detected by CleanMail (false negatives), and some are tagged as spam even when they aren't (false positives). Learning messages, in short, is about teaching CleanMail to do better for similar messages in the future.

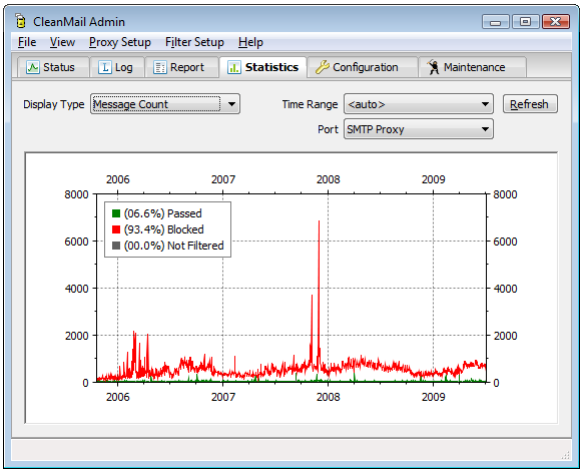

Figure 5.2: Statistics View

There are two filter types supporting this type of learning: **Fingerprint** filters, and SpamAssassin filters.

Fingerprint filters use a proprietary technique to create one or more short hash codes (fingerprints) of a message to summarize its structure and content. Known spam message fingerprints are stored in a file. You can explicitly add fingerprints of spam messages to this file.

SpamAssassin stores the relationships of words in a small database (the SpamAssassin Bayes database). Words stored in this database may increase or decrease the total spam score of a message. You can add words to this database by learning a messages either as ham or as spam.

Learning messages requires the use of a mail storage filter ((see *Mail Storage Setup* (section 4.11)). To learn a message, locate the message in the *Journal Report* (section 5.3) of the admin application. Right-clicking the message opens a menu that allows learning messages:

Learn As Ham: The message is learned as a ham message.

Learn As Spam: The message is learned as a spam message.

Forget: All database entries already learned from a message are deleted. If you accidentally learned a spam message as ham (or spam), you can undo it this way.

Depending on how you configured the the mail storage filter, a message file may have been deleted already when you try to learn it as either spam or ham. To avoid this in the future, change your mail storage settings to keep a longer backlog of older messages.

Similarly, messages can also be learned from the **Journal** view of the Web Dashboard.

NOTE: The results of the Bayes tests are ignored by SpamAssassin until at least 200 messages have been learned.

CAUTION: Never learn forwarded spam mails. The act of forwarding modifies mails in unexpected ways, learning a forwarded mail will be useless or even counter-productive.

You can also train the SpamAssassin Bayes database from a command window by using `sa-learn`. For more information on this procedure, see *Using Sa-learn in a Command Window* (section6.3.3).

5.7 Using Blacklists and Whitelists

Blacklist and Whitelist filters can be used to permanently block specific sender addresses (blacklist), or to permanently allow messages from specific sender addresses to bypass all filtering (whitelist).

Addresses can be added to both the whitelist and the blacklist very comfortably from the **Report** page of the CleanMail Admin application. While the whitelist has proven to be very useful, however, but the blacklist rarely is. Often senders of spam or virus messages do not use a particular sender address mor than once, so blacklists are regularly and easily bypassed. If you freely add mail addresses to your blacklist, it will soon get large, unwieldy, and ever more difficult to maintain. So, whenever you add an address to the blacklist, ask yourself these questions:

- Is it really likely that I will receive another message with the same address? When in doubt, it is better to learn this message as spam, so that its fingerprints can be identified when you receive it next time with a different sender address.
- Is it more efficient to block messages from an entire domain, using wildcards, like in `*@obnoxious.com`?

5.8 Tuning The CleanMail Filter Pipeline

A central part of the CleanMail configuration is the tuning and optimization of the filter pipeline. As each filter analyzes a message in turn, it uses resources such as CPU processing power or memory. Obviously, the order of the filters in the pipeline matters. If the first filter is known to consume lots of resources overall server throughput will be reduced. On the other end of the spectrum, light-weight filters may be used as a ”triage” stage: obvious spam is discarded, freeing precious server

resources, possibly at the cost of a higher probability of classifying legitimate mails as spam (false positives). However, resource usage is only one aspect of a much more complex issue, there are other criteria to look at when configuring your filters. Here's a list:

Aggressivity This term describes the likelihood of a filter discarding a legitimate message, resulting in a so called false positive. Usually you do not want to get false positives, but unfortunately aggressive filters often execute very fast with little resource usage. The aggressivity of some filters can be configured, but configuring a filter to be less aggressive also increases the likelihood of spam messages passing through (false negatives).

Resource Usage Filters performing complex tests, and providing a high degree of flexibility usually also require a large amount of system resources, such as memory or raw computing power. Depending on the amount of mail you want to filter, this may not be an issue at all, but if it is, you should avoid running these filters for each and every message you get.

Selectivity Filters able to classify only a small percentage of messages as definitely legitimate or definitely not legitimate, passing a large percentage of "undecided" messages to the following filters are said to have a low selectivity. For example, it may not be worthwhile to run a resource-consuming filter, if its selectivity is very low.

Type Malicious message fall into two categories: spam messages (including scams and phishing attacks), and virus messages (containing and propagating worms, trojans and viruses). Some filters are effective against spam only, others are effective on virus messages only, and some are effective against both.

CleanMail message filters can be classified according to the following table:

Filter	Aggressivity	Resource Usage	Selectivity	Protection
Anti-Virus	low	high	medium	anti-virus
Attachment	low	low	medium	anti-virus
SMTP Delay	low	low	medium	both
DNSBL	high	low	high	both
Fingerprint	medium	low	high	both
SMTP Checks	low	low	medium	both
SpamAssassin	low	high	medium	anti-spam

The built-in SMTP-level filtering (traffic limiting, anti-abuse), is not configurable in filter pipeline, and is only available in SMTP proxies.

CleanMail by default orders the filters to optimize throughput, using the following guidelines:

- Filters with the lowest resource usage and the highest selectiveness go first. For this reason the fingerprint filter is always be one of the first filters in the filter pipeline, because of its low resource usage and its good results in blocking spam and malware.
- Filters which use a lot of processing power and with low selectiveness go last. Therefore, SpamAssassin is one of the last filters. It does a good job at detecting spam, but its CPU and memory usage may prohibit its use for every message received.

5.8.1 Choosing the Right Filters

Judging from the list above, the following filters are a must-have in every CleanMail configuration, in order of their execution:

Attachment Blocker A no-brainer, with a static set of blocked attachments, this filter gets rid of many virus messages at practically no cost.

Fingerprint Filter This filter gets rid of spam and virus messages without using up resources. Though long-term studies are still unavailable, the additional risk of false positives appears to be very small.

SMTP Delay Another no-brainer. This filter in its simplicity makes you wonder, why it hasn't been countered by spammers yet. Sometimes legitimate batch mailers run into problems with this, if they have set their timeouts set too low. In this case, remove this filter.

SpamAssassin A classic. The leading open-source spam filter, highly flexible with exceptionally good results, though at the cost of heavy resource usage.

Anti-Virus You can use the open source Clam AV scanner, or integrate any other third party scanner. Use multiple virus scanners, if you have the necessary processing power available.

The other filters are situational - your milage may vary:

Blacklist There are situations where this filter is useful, but in general it is largely ineffective, as spammers usually use a different fake address for every message.

DNSBL DNSBLs sometimes are too aggressive, but overall the low resource usage of these filters may help you out of a tight spot if your filtering server runs into system load trouble. If not, there is no need to add this filter, as SpamAssassin already integrates DNSBLs in a less aggressive form. The effectiveness of DNSBLs for IPv6 is in doubt, given the fact that a spammer

can use a different IP address for every message he sends. At the moment only a small percentage of spam and viruses delivered using IPv6, so this is not an issue (yet).

Whitelist Use if needed. You can configure for every filter individually if the whitelist should be ignored, as some users prefer to run anti-virus and attachment filtering even for messages originating from whitelisted senders.

When configuring CleanMail with the Admin application, every new filter will be automatically moved to the best position in the filter pipeline. Afterwards, you can still change the order of filters, but only within limits.

5.8.2 Example Filtering Results

The figure below shows typical filtering results for a CleanMail filter pipeline, using an attachment blocker, the fingerprint filter, the delay filter, SpamAssassin and Clam Anti-Virus, in that order, with the built-in SMTP checks as an added bonus filter getting rid of abusive SMTP traffic even before the filter pipeline is invoked.

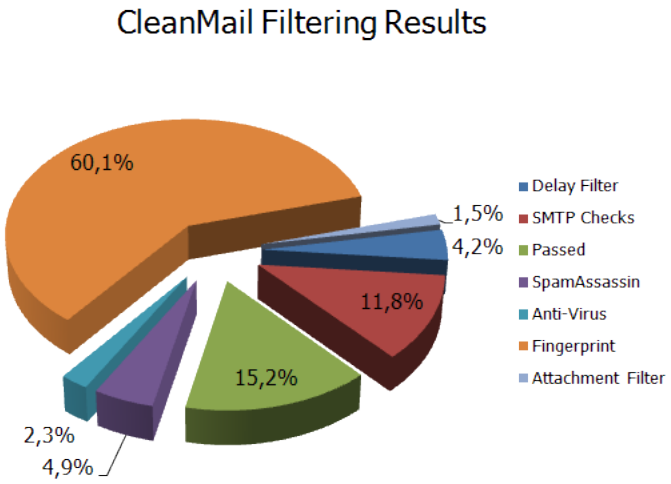

Figure 5.3: CleanMail Filtering Results

The low-resource usage filters are able to discard 77.6% of all incoming mail traffic, before the rest (22.4%) is passed to the more elaborate filters such as SpamAssassin and Anti-Virus, finally leaving only 15.2% of all messages classified as legitimate and passed on to the recipients. The fingerprint filter in this example, being one of the first filters, is able to get rid of the lion's share of all unwanted messages, removing this filter would increase the slices for SpamAssassin

and Anti-Virus proportionately. In summary, the light-weight filters are able to increase your CleanMail server's message throughput almost five-fold, in comparison to a solution only using SpamAssassin and Anti-Virus.

Looking at the chart and the numbers, you might be misled into thinking that SpamAssassin and Anti-Virus have a rather small effect on the filtering results and can be removed from the filter pipeline with only little impact on results. However, precisely these filters are needed to teach spam fingerprints to your fingerprint database, as a spam message has to be filtered at least once by some other filter, before all subsequent occurrences of similar messages can be discarded by the fingerprint filter. After removing SpamAssassin and Anti-Virus from the pipeline, the fingerprint filter would effectively stop to work.

5.8.3 Troubleshooting

Too many false positives Chances are you are using one of the more aggressive mail filters. Get rid of DNSBL, if the processing power of your server permits. Always learn false positives as ham to improve future results, and whitelist any senders that are repeatedly blocked by your filtering. Make searches available to your users on your Intranet, so they can check the list of blocked messages themselves.

Too many false negatives Add more filters. With every new filter the chances increase that a particular spam or virus message could have been detected. Learn false negatives as as spam.

Server is at 100% CPU constantly Check for mail flooding. If you are not being flooded, and the high resource usage is constant, add filters with low resource usage and high selectivity. The fingerprint filter is a must, and you may also need DNSBL. If all of this does not help, upgrade your hardware.

5.9 Web Dashboard

You can also monitor your CleanMail installation using a web browser. Use this URL to access the monitoring page served by the CleanMail service while it is running: http://localhost:8086/index.html.

The web dashboard allows access to almost all of the monitoring and reporting functions.

Chapter 6

Reference

6.1 CleanMail Configuration File

CleanMail's configuration is saved in a plain text file, `cleanmail.cf`. For easy configuration, you can access the configuration settings using the cleanmail admin application, but advanced users can also use a simple text editor to change the settings by modifying the file directly.

Note: To make CleanMail re-read its configuration file, simply restart the service, using either the Windows services manager, or the admin application (choose 'Apply Settings' from the file menu). Likewise, the admin application only re-reads the configuration file upon restart.

Note: Some configuration files originated on other operating systems than Windows, and the editor supplied with Windows (`notepad` may be **unsuitable** for editing these files. Install some other text editor package available in the Internet, such as `crimson`, `textpad` or `ultraedit`, just to name a few.

6.1.1 General Structure

The configuration file format is similar to the Windows `*.ini` file format.

- The first line identifies the file format version. At present, only file format versions 3.0, 2.1, and legacy files (CleanMail version 1.x) are supported.
- Configuration settings are grouped in sections. Each section starts with a section label in square brackets (`[]`).
- A section may include other sections (subsections). The section label of such a section repeats the section label of the enclosing section, and its own label.

- Configuration settings are given as `<name>="<value>"`. Note that values should always be quoted (using double-quote characters (`"`). Empty values (denoted by a pair of double quotes with no other characters between) may be allowable for some settings.
- Lines starting with the `#` character are ignored, and can be used to add comments.

All settings not explicitly overridden in the configuration file, take on their default value.

6.1.2 Value Types

Configuration settings can have the following types:

boolean Set to either 0 (meaning false, no, disable), or to 1 (meaning true, yes, enable). In some cases it may be allowable that a boolean value may be empty or unset (meaning unset, undefined, unknown).

numeric Set to a numerical value. Numbers must be given as classic decimal numbers, a leading − denotes negative numbers, the dot character `.` is used as decimal point. In some cases it may be allowable that a numeric value is unset (empty).

string A sequence of printable US-ASCII characters, other than the double quote character.

6.1.3 Session Manager Settings

The first section of the config file is a session manager section, labelled `[WorkstationSessionManager]`.

CheckForUpdates (boolean) Enables or disables the check for program updates at midnight. This value defaults to `"1"` (enabled).

CheckForUpdatesProxy (string) Sets the HTTP proxy server and port used for the update check. Defaults to an empty value, in this case CleanMail connects directly to `www.byteplant.com`, port 80, to perform the update check.

DetailedLogging (numeric) Sets the logging level of cleanmail. Individual bits in the binary representation of this number enable different logging options. Defaults to `"16"`:

1 Detailed logging

2 Filter error output

16 Extended logging

DNSServer (string) List of DNS servers to use. Defaults to an empty value. in this case CleanMail tries to determine the DNS servers automatically.

MaxBufferSize (numeric) The maximum message size in Byte, larger messages are rejected. This is an important security feature, if you set this too large, users of your mail service can crash your server by submitting a very large message. Allowable values are 10485760 (10MB) to 1048576000 (1000MB), the default is 20971520 (20MB).

MaxLogFileDays (numeric) The log file is cycled after so many days. Set to `"1"` by default (daily log file cycling), both unsetting this value or setting it to 0 disables this feature.

MaxLogFileSize (numeric) The log file is cycled once it is larger than the given size. Allowable values are 1048576 (1MB) to 1048576000 (1000MB), the default is unset, disabling this feature.

MaxMemoryUsage (numeric) A load factor determining cleanmail's memory usage. The default value is 0.8, allowable values are in the range between 0 to 1.

PreferEnableFilter (boolean) Determines cleanmail's filter behaviour if a mail is addressed to multiple recipients. Defaults to "true", in this case a filter is applied if the filter is enabled for at least one recipient, and not applied otherwise. If set to false, a filter is not applied, if the filter is disabled for at least one recipient, and applied otherwise.

StatisticsTimeFrame (boolean) Sets the number of days how long statistics data is kept. Defaults to 365, allowable values are in the range of 7 (one week) to 10000 (about 30 years).

TemporaryDirectory (string) Directory used to store temporary files. Leave this empty to use the default temporary directory of the account (as set in the TMP and TEMP environment variables).

6.1.4 Port Settings

The session manager section may contain multiple proxy port subsections. CleanMail supports different types of proxy ports, with the port type determined by the section label. A POP3 port, for example, has the following section label: `[ServerSessionManager\Ports\POP3Port]`. All types of proxy ports have several settings in common.

6.1.4.1 General Proxy Port Settings

These settings apply to all types of ports.

IncomingConnectionCount (numeric) Sets the length of the listening queue on the proxy port server socket. Defaults to 20, allowable values are operating system dependent.

IncomingPort (numeric) Sets the port number of the server socket. The default depends on the protocol involved. Allowable values are in the range of 0 to 65535.

IncomingServer (string) Sets the IP address or name of the server socket. If empty, the socket is bound to all interfaces. Defaults to an empty value.

IncomingTimeout (numeric) Sets a timeout, in seconds, for waiting on data from the client connected. Upon timeout, the incoming connection is reset and resources held by the connection are freed. The default depends on the protocol involved. Allowable values are in the range of 10 to 3600.

Name (string) Port name, used for housekeeping purposes. Port names should be unique. The default depends on the protocol involved.

OutgoingConnectionCount (numeric) Set the maximum number of simultaneous active connections on this port. Defaults to 1000, allowable values are in the range of 1 to 1000.

OutgoingTimeout (numeric) Sets a timeout, in seconds, for waiting on data from the server connected. Upon timeout, the incoming connection is reset and resources held by the connection are freed. The default depends on the protocol involved. Allowable values are in the range of 10 to 3600.

6.1.4.2 HTTP Port Settings

Monitoring ports (HTTP ports) only support the general port settings, and the following additional settings:

Allow (string) A list of IP addresses or hostnames that may access the monitoring port. All addresses within private network IP address blocks (RFC-1918) are automatically allowed. If you want client access over the Internet, enter the remote hosts that may access the server. Note that dynamic DNS is supported. Defaults to an empty value.

Deny (string) A list of IP addresses or hostnames that may not access the monitoring port. Defaults to an empty value.

Password (string) Password required to access monitoring and admin data. If both user and password are set, the web browser will prompt for a password if you try to access the reporting pages, and you also need to enter the user name and password in the CleanMail admin application's connection settings (choose 'Connect...' from the 'File' menu).

User (string) User name required to access monitoring and admin data. If both user and password are set, the web browser will prompt for a password if you try to access the reporting pages, and you also need to enter the user name and password in the CleanMail admin application's connection settings (choose 'Connect...' from the 'File' menu).

UseSSL (boolean) Enables SSL for the HTTP port. This will only work if both a certificate file (`CleanMail.cert`) and a private key file (`CleanMail.key`) are present in the configuration directory. Read permissions of the key file must be restricted to the account running Cleanmail. Check the log after startup if your certificate has been accepted. This setting defaults to false.

6.1.4.3 POP3 Port Settings

POP3 ports support the general port settings. Also, POP3 ports may contain multiple filter sections, describing the filters configured for this port, see *filter settings* (section6.1.5).

6.1.5 Filter Settings

6.1.5.1 General Filter Settings

ID (string) Unique, short filter identifier used internally. A unique ID is automatically generated by default. Allowable identifiers are not empty, and have up to 7 characters.

IgnoreWhitelist (boolean) If true, the filter is always applied, even to whitelisted messages. Defaults to `false` for most filters, with the exception of attachment filters and anti-virus filters.

Name (string) Filter name, used in statistics and log files. Filter names should be unique. A unique name is usually generated by default. Allowable identifiers are not empty, and up to 30 characters.

SubjectTag (string) If not empty, the message subject is modified to indicate that a message is junk. This setting allows `US-ASCII` non-control characters only (character codes 32-127). The default setting is "*SPAM*". **Note:** This setting has no effect for the SpamAssassin filter. Change the local.cf file instead (see SpamAssassin documentation).

6.1.5.2 Attachment Filter Settings

Attachment filter settings are defined in an AttachmentConfig section `[(...)\Filters\AttachmentConfig]`. Attachment filters support the following additional configuration options:

BlockList (string) Sets the attachment types you want to block (deliver the message with the attachment(s) removed). You can use the wildcard characters `?` (any character) and `*` (any number of any character) to define attachment types. Multiple attachment types can be separated by blanks. If empty, no attachments are blocked. The block list is the first attachment type list checked, attachments that match a type pattern on this list are not checked against the drop list. Defaults to a list of attachment types that may carry macro viruses.

BlockListPolicy (string) Filter action applied to messages containing attachment(s) of a blocked type. Defaults to `reject/deliver`. Allowable values are: `accept/deliver`, `reject/deliver`. The attachment is always removed.

DropList (string) Sets the attachment types where you want to delete the entire message if it contains a restricted attachment. You can use the wildcard characters `?` (any character) and `*` (any number of any character) to define attachment types. Multiple attachment types can be separated by blanks. If empty, no messages are deleted. The block list is the last attachment type list checked. Defaults to a list of attachment types commonly used in virus messages.

DropListPolicy (string) Filter action applied to messages containing attachment(s) of a blocked type. Defaults to `reject/delete`. Allowable values are (in order of precedence): `reject/redirect`, `accept/redirect`, `reject/delete`, `accept/delete`.

ErrorLineBreak (boolean) If this is set to true, bad line breaks are treated like other MIME errors, and handled as defined in the ErrorPolicy setting. Defaults to `false`.

ErrorPolicy (string) Filter action applied to messages with unrecoverable MIME syntax violations. This usually involves malformed attachment specifications, or other techniques that could be exploited to bypass the attachment filter or virus checkers. Defaults to `accept/deliver`. Allowable values are (in order of precedence): `accept/deliver`, `reject/redirect`, `accept/redirect`, `reject/delete`, `accept/delete`.

PassList (string) Sets the attachment types you want to always accept. You can use the wildcard characters `?` (any character) and `*` (any number of any character) to define attachment types. Multiple attachment types can be separated

by blanks. The pass list is the first attachment type list checked, attachments that match a type pattern on this list are not checked against the block list or the drop list. Defaults to empty.

6.1.5.3 Blacklist and Whitelist Filter Settings

Blacklist filter settings are defined in a BlacklistConfig section `[(...)\Filters\BlacklistConfig]`, whitelist filters in a WhitelistConfig section `[(...)\Filters\WhitelistConfig]`. The following additional configuration options are supported:

SenderList (string) List of sender address patterns. You can use the wildcard characters `?` (any character) and `*` (any number of any character). Multiple addresses are separated by blanks.

Policy (string) Filter action applied to messages that have a sender address matching a pattern in the sender list. Defaults to `reject/delete` for a blacklist filter, and to `accept/deliver (whitelisted)` for a whitelist filter.

6.1.5.4 RBL Filter Settings

RBL filter settings are defined in a DNSBLConfig section `[(...)\Filters\DNSBLConfig]`. The following additional configuration options are supported:

ZoneList (string) List of DNSBL zones to query. Multiple zones are separated by blanks.

Policy (string) Filter action applied to blocked messages. Defaults to `reject/delete`.

RelayCheck (string) Defines which received headers (relays) are checked by the filter. If DNS blacklists contain dial-up services, you can reduce the risk of false positives by skipping the DNSBL check for the first received header (created when a dial-up sender submits his message to the first SMTP server). Note that the last received header (last relay) will always be checked, even when there is only one received header. Allowable values are `all`, `all but first`, and `last only`. The default setting is `all but first`.

Timeout (numeric) When the timeout is exceeded, all DNS queries are cancelled and the message is forwarded to the next filter (Result: 'unknown'). Recommended values are in the range of 15-30 seconds, the default is 30 seconds.

6.1.5.5 Shared Real-Time Fingerprint Filter Settings

Shared Real-Time Fingerprint Filter settings are defined in a FingerPrintConfig section `[(...)\Filters\FingerPrintConfig]`. The following additional configuration options are supported:

Policy (string) Filter action applied to blocked messages. Defaults to `reject/delete`.

MaxFilterSize (numeric) Spam and virus messages are usually small. To conserve system resources and increase throughput, it is recommended to skip filtering mails exceeding a certain size, given in Byte. Allowable values are in the range of 1024 (1kB) to 1048576000 (1000MB). Defaults to 524288 (512kB).

6.1.5.6 External Filter Settings

These settings can be used with all filters based on this filter type. External filters may also be used by themselves, in this case the settings are defined in a CommandLineConfig section (`[(...)\Filters\CommandLineConfig]`. External filters support the following additional configuration options:

CommandLine (string) Sets the command line to execute this filter. You can use `%IN%`, `%OUT%`, or `%ERR%` as placeholders for the input, output and error file names. If `%IN%` is not used, the input message is available as standard input, if `%OUT%` is not used, output is collected from standard output, if `%ERR%` is not used, error output is collected from standard error. The defaults depend on the filter implementation.

CommandLineOutput (boolean) Some filters do not modify a message. If you unset this option the filter input is forwarded to the next filter in queue, and any filter output other than the exit code is ignored. Defaults to true in plain command line filters, but this default is overridden in other filters based on the command line filter.

CommandLinePriority (numeric) Sets the scheduling priority of the process executed. On Linux/Mac this setting corresponds to the process nice value, and defaults to `"0"` (normal priority). On Windows, allowable values are`"0"` for low priority, and `"1"` for normal priority (the default).

CommandLineTimeout (numeric) Sets a timeout, in seconds. If the timeout is exceeded the message is accepted without changes (Result: 'unknown'). Allowable values are in the range of 10-1000s. Defaults to 60s in plain command line filters, but this default is overridden in other filters based on the command line filter.

MaxFilterSize (numeric) Spam and virus messages are usually small. To conserve system resources and increase throughput, it is recommended to skip filtering mails exceeding a certain size, given in Byte. Allowable values are in the range of 1024 (1kB) to 1048576000 (1000MB). Defaults to empty in plain command line filters, but this default is overridden in other filters based on the command line filter.

MaxMemoryRequired (numeric) Specifies how much system RAM the filtering program needs. This setting helps CleanMail to optimize resource allocation. Allowable values are in the range of 1048576 (1MB) to 1048576000 (1000MB). Defaults to empty in plain command line filters, but this default is overridden in other filters based on the command line filter.

UseDOSPathNames (boolean) Set to true if the filter is a DOS program that requires DOS 8.3 file names on the commandline. This setting has no effect under operating systems other than Windows.

Commandline filters may contain multiple return code sections.

6.1.5.7 Return Code Settings

Return codes are defined in ReturnCode sections: `[(...)\Filters\CommandLineConfig\ReturnCodes\ReturnCode]`

Code (numeric) The program exit code. Allowable values are in the range of 0 to 255.

Policy (string) Sets the filter action to apply if the command line program returns the exit code defined in the Code setting. Allowable values are: `accept/deliver` `accept/deliver (junk)`, `reject/deliver`, `reject/redirect`, `accept/redirect`, `reject/delete`, `accept/delete`.

6.1.5.8 Mail Storage Settings

Mail storage filter settings are defined in a CacheConfig section.

CacheDirectory (string) Specifies the directory where messages are stored. Defaults to empty. If empty, the filter will store files in the temporary directory.

MaxCacheDays (numeric) Sets the maximum number of days messages are kept in storage. Values can be in the range of 1 to 31 days. Defaults to unset.

MaxCacheFiles (numeric) Sets the number of messages to store in the cache directory. Once the limit is reached, for every new message stored, the oldest message is deleted.' Allowable values depend on the file system used. Defaults to 1000.

MaxCacheSize (numeric) Sets the maximum disk space used in KByte (note that values are displayed as MByte in the admin application). Allowable values are in the range of 102,400 (100MB) to to 10,485,760 (10GB). Defaults to unset.

6.1.5.9 Antivirus Filter Settings

Anti virus filter settings are defined in a AntiVirusConfig section. In addition to the general filter settings and the command line filter settings, the anti virus filter supports the following settings:

CommandLine (string) In difference to plain command line filters, antivirus filters substitute the `%SCANNER%` placeholder with the executable defined in the scanner setting. The default setting is operating system dependent, and vendor dependent.

Scanner (string) Specifies the complete path and file name of the command line scanner executable. Defaults to empty.

VendorName (string) The name of the anti virus scanner vendor. Defaults to empty. Allowable names are operating system dependent. This setting only affects the CleanMail admin application.

6.1.5.10 SpamAssassin Filter Settings

SpamAssassin filter settings are defined in a SpamAssassinConfig section. In addition to the general filter settings and the command line filter settings, the SpamAssassin filter supports the following settings:

CommandLine (string) In difference to plain command line filters, SpamAssassin filters substitute the `%CONFIGPATH%` placeholder with the ruleset path defined in the SpamAssassinRulesetPath setting. Also the `%FLAGS%` placeholder is substituted with runtime flags and should not be omitted. The default setting is `sa\spamassassin.exe %FLAGS% -x -c \"%CONFIGPATH%\" -e 255`.

DropThreshold (numeric) If you set a DropThreshold, mails are deleted if the spam score is higher than this value, regardless of the return code policy settings. Allowable values are in the range of 3.0 to 1000.0, the default is empty. If empty, this feature is disabled.

SpamAssassinRulesetPath (string) Specifies the complete path and of the SpamAssassin ruleset directory. The default is operating system dependent.

CommandLineOutput (boolean) Defaults to false, see section *Command Line Filter Settings* (section6.1.5.6) for more information.

CommandLineTimeout (numeric) Defaults to unset (disabled), see section *Command Line Filter Settings* (section6.1.5.6) for more information.

MaxFilterSize (numeric) Defaults to 524288 (512kB), see section *Command Line Filter Settings* (section6.1.5.6) for more information.

MaxMemoryRequired (numeric) Defaults to 268435456 (25MB), see section *Command Line Filter Settings* (section6.1.5.6) for more information.

SpamAssassin filters define two return codes: one for exit code 0 (non-spam or ham message), and for exit code 255 (spam message). As ham message policy `accept/deliver` is recommended, and as spam policy one oft the following should be chosen: `accept/deliver (junk)`, `reject/deliver`, `reject/redirect`, `accept/redirect`, `reject/delete`, `accept/delete`).

6.1.6 Search Settings

The SessionManager section of the config file may contain one or more CacheSearch sections, holding search definitions. The settings are:

CacheName (string) The name of the mail storage to search in. Set to a combination of the mail storage name followed by the proxy port name in parentheses. A typical example could be `Mail Storage (SMTP Proxy)` (using the default mail storage name and the default SMTP proxy port name). The search will fail if no mail storage with this name exists.

Name (string) The name of the search, used to identify a search. The search name must be unique.

Password (string) Password required to access the search using the web interface. If both user and password are set, the web browser will prompt for a password if you try to access the reporting pages. The global allow/deny hosts settings for the HTTP port always apply.

User (string) User name required to access monitoring and admin data. If both user and password are set, the web browser will prompt for a password if you try to access the reporting pages. The global allow/deny hosts settings for the HTTP port always apply.

Message in the cache can be filtered using search terms, which basically are name/value pairs, with the name defining an envelope field or message header field to scan, and the value a pattern matched against the content field of this header field. Search patterns support the wildcard characters `?` (any character) and `*` (any number of any character), for example like in `*@byteplant.com`.

A search without any search terms will return all message in the mail storage, a message with search terms returns all messages matching with at least one search term.

For performance reasons, only a subset of header fields can be searched:

MessageID The X-CleanMail message ID assigned by CleanMail. This can be used to find a particular message in the cache.

Action The filter action taken for a message.

From The mail address extracted out of the 'From' header field of a message. The from field often does not match the from address used in during transport.

HostIP The IP address of your ISP's POP3 server.

HostName The name of your ISP's POP3 server.

Policy The filter policy returned by the filtering pipeline.

POP3Account POP3 account name, in the form user@host.

Port Matches against the port name that transported this message.

Subject The message subject. Searches against the subject may not match if the subject contains special characters, or unsupported character encodings.

To The mail address(es) extracted out of the 'To' header field of a message. The to field often does not match the recipient address used in during transport.

6.2 Log Files

The location of the log files depends on the operating system version and the operating system language. The main log file us called `CleanMail.log`.

Statistics data is kept in a file called `CleanMail_Statistics.csv`. The statistics file is a list of comma separated values, and thus can be easily read and processed by spread sheet software such as Microsoft Excel.

The data is organized in lines, each line the counter values for a day. The first field of a line contains the date.

The statistics file is cycled every day at midnight, or whenever the service is stopped. At the same time today's values will be added or updated.

Once the date of the first line is older than 365 days, CleanMail will delete this line. This way the statistics file keeps data for up to one year back.

Statistics counters are identified by descriptive names. The data collected includes the following:

- Totals counted for a proxy port (mails filtered, mails passed, mails blocked, total, traffic received/sent on the incoming/outgoing ports)
- Filter results for different filters, counted separately for each proxy port.
- The number of attachments blocked by the attachment blocker, and the number of mails were MIME violations were detected (only if attachment filter is used).

For every mail received or rejected a line is added in CleanMail's mail log file (`CleanMail_mail.csv`). The mail log file is a list of comma separated values. Like the statistics file it can be read and processed by spread sheet software such as Microsoft Excel®[1].

6.3 SpamAssassin

The Windows release of CleanMail is bundled with a ready-to-run SpamAssassin installation. All related files are located in the sa subdirectory of the CleanMail installation path, including all executable files and scripts, and the configuration files.

6.3.1 SpamAssassin Main Configuration Files

SpamAssassin's main configuration files can be found in the `sa\ruleset` folder:

```
local.cf

user_prefs

init.pre

v310.pre

v312.pre
```

[1] Microsoft Excel is a registered trademark of Microsoft Corporation

```
v320.pre

v330.pre

sa-updatechannels.txt
```

All configuration changes in these files are preserved during updates, whereas other ruleset files are likely to change, so do not edit any other files.

6.3.2 SpamAssassin Ruleset Updates

CleanMail automatically updates the SpamAssassin rules and scores in the `sa\share\ruleset` folder every 24 hours by running `sa-update`. The update channels (download locations) can be configured in the `sa-updatechannels.txt` file (located in the `sa` folder).

You can also manually update the SpamAssassin ruleset by running the `sa-update.bat` batch file.

6.3.3 Using Sa-learn in a Command Window

Using `sa-learn` directly allows learning multiple messages or entire mail folders at once, and it gives you more flexibility to adapt your CleanMail installation to your environment. However, you must be sure that messages are either available as an ASCII file in its raw MIME format (RFC-2822), or in the `mbox` format. Many popular mail software packages such as Mozilla Thunderbird support these formats, whereas some, like all Microsoft products, do not.

The documentation of `sa-learn` can be found in the `sa\doc` subdirectory or online. However, the ”official” documentation has to be taken with a grain of salt, as it hasn't been written with Windows as a target operating system in mind. Most important: Beware of blanks in pathnames. Be sure to use double quotes if a path or file name contains blanks!

CAUTION: Never learn forwarded spam mails. The act of forwarding modifies mails in unexpected ways; to learn a forwarded mail will be useless or even counter-productive.

Learn single messages

Find the X-CleanMail-MessageID header in the message you want to learn. Using the MessageID you can locate the `.eml`-file in the mail storage directory. To learn a message as spam from the command window, use the following command line:

```
cd [InstallationDirectory]
sa\sa-learn --siteconfigpath="sa\ruleset" --spam "[Path]"
```

To forget or learn as ham, use `--forget` or `--ham` instead of the `--spam` option.

Learn a message folder

Most mail clients are using the mbox mail folder format or have an export function to export a mail folder to an mbox file. Collect the spam messages you want to learn in a mail folder and export this folder to an mbox file. Then use the following commands in a command line window:

```
cd [InstallationDirectory]
sa\sa-learn --siteconfigpath="sa\ruleset" --spam --mbox "[Path]"
```

For repeated use, create a batch file with the commands above. An example is provided in the installation directory of CleanMail.

If you are using Microsoft Outlook™or Outlook Express™[2], you can't learn entire mail folders, because there is no simple way to export to an mbox file. There is not even a way to export a single message to a text file in RFC-822 format. (There are some tools around, look for `outlook2mbox` or similar with your favorite Internet search engine, but your mileage may vary.)

However, there is a way, even if you are using Microsoft Exchange as mail server. This requires the administrator to install one email client other than Outlook or Outlook Express. This mail client can then be used to fetch the mails to learn by POP3 or IMAP. A step-by-step example, using Mozilla Thunderbird, is described on this web page:.

6.3.4 SpamAssassin Database Expiry

As SpamAssassin continues to learn from spam and ham mails, its Bayes database continues to grow. CleanMail regularly checks if the SpamAssassin database exceeds a certain size limit (100,000 tokens or words, about 5MB in database size). Once the limit is reached, old tokens (words that were not encountered in mails for a long time) are removed from the database. This is called database expiry.

On slow systems, it can happen that a mail transmission is stalled for several minutes while the database expiry is underway. CleanMail tries to pick an expire time where no transmission is in progress (typically during the night).

Normally, SpamAssassin database maintenance does not require any user interaction. You can use sa-learn to examine the state of the SpamAssassin database from time to time, or to manually force a database expiry. Please check the sa-learn documentation (located in the `sa\doc` subdirectory).

[2]Microsoft, Microsoft Outlook and Microsoft Outlook Express are (registered) trademarks of Microsoft Corporation

6.4 POP3 command quick reference

POP3 is readable by humans. This section provides the necessary knowledge to interpret the information given in cleanmail's logs (if you enable detailed logging) and understand what happens during a POP3 mail retrieval session. The complete specification of the POP3 protocol is given in the RFC-1939 document, and available online: http://www.ietf.org/rfc/rfc1939.txt.

6.4.1 Example POP3 Session

For testing purposes, you can also retrieve mail by means of a text-only terminal session, e.g. using telnet to connect to the POP3 port of a POP3 server.

Once the connection has been opened, the POP3 server issues a one line greeting. Here's an example:

```
+OK pop3.byteplant.com ready
```

Now is the time for the client to sign in, issuing the `USER` and `PASS` commands:

```
USER test
+OK
PASS fred
+OK
```

The `LIST` command can now be used to get a listing of all messages stored in the mail box.

```
LIST
+OK 2 messages
1 344
2 857
.
```

In the example, two messages are in the mailbox, numbered 1 and 2, with lengths of 344 and 857 byte.

The `RETR` command now fetches the message, the argument is the number of the message to be fetched:

```
RETR 1
+OK 344 octets
Received: from [127.0.0.1] ...
From: god@heaven
To: mortal@earth
```

```
Subject: Test Message

This is the body of the test message
.
```

Fetching a message with the `RETR` command does not delete this message, it still remains in the mailbox. For deleting a message, the `DELE` command is used:

```
DELE 1
+OK message 1 deleted
```

However, POP3 servers do not actually delete messages until the session has been closed. Interrupting a POP3 session for this reason may cause messages to be transmitted to the POP3 clients multiple times. So in order to make deleting the message permanent, the session has to be closed using the `QUIT` command.

```
QUIT
+OK Bye
Connection closed by foreign host.
```

6.4.2 POP3 commands

USER Specify a mailbox or user name

PASS The password for the mailbox.

LIST Lists messages in the mailbox

RETR Fetch a particlar mail from the mailbox

DELE Mark a message for deletion

QUIT End POP3 session, delete messages marked for deletion.

6.4.3 Server replies

POP3 server replies begin with a +OK, if no error occurred, or with -ERR in case of an error.

Chapter 7

Licensing and Contact Information

7.1 Ordering CleanMail

For the latest pricing information, please visit our online shop.

CleanMail is distributed online electronically and shipped on CD-ROM, if requested. Please visit our online shop to place your order online. Ordering online and paying by credit card is by far the fastest way to order: Your license key is usually delivered in a matter of minutes.

If you do not want to order online using your credit card, we offer a variety of alternative ordering methods. Please visit our online shop to find out more.

7.2 Support

A purchase of CleanMail includes free email support for at least 6 months. Please write to us at support@byteplant.com. We will also try to help you with the Trial version of CleanMail if we can.

Contact us for information regarding other support options by email to sales@byteplant.com

For the latest version always check the CleanMail download page.

Byteplant offers consulting and the development of custom software. Please inquire by email to sales@byteplant.com.

7.3 Copyright

Byteplant GmbH

Heilsbronner Strasse 4

D-91564 Neuendettelsau / Germany

E-Mail: contact@byteplant.com

Company Homepage: http://www.byteplant.com

7.4 License and Usage Terms

END-USER LICENSE AGREEMENT FOR CLEANMAIL This End-User License Agreement (”EULA”) is a legal agreement between you (either an individual or a single entity) and byteplant GmbH. If you do not agree to the terms of this EULA, do not install, copy, or use CleanMail.

SOFTWARE PRODUCT LICENSE CleanMail is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. CleanMail is licensed, not sold.

LICENSE/USAGE TERMS For payment of the license fee the licensee is granted one (1) non-exclusive, non-transferable license to install and use CleanMail Home on one (1) computer at a time or install CleanMail Home on one (1) computer to be used by multiple users. It is expressly forbidden to install CleanMail Home for use on multiple computers without paying additional license fees. Licensee warrants that they will make a reasonable effort to remove unused licenses of CleanMail Home. Please contact us via email at support@byteplant.com for site licenses and volume discounts.

DISCLAIMER OF WARRANTY CLEANMAIL AND THE ACCOMPANYING FILES ARE SOLD ”AS IS”. BYTEPLANT MAKES AND CUSTOMER RECEIVES FROM BYTEPLANT NO EXPRESS OR IMPLIED WARRANTIES OF ANY KIND WITH RESPECT TO THE SOFTWARE PRODUCT, DOCUMENTATION, MAINTENANCE SERVICES, THIRD PARTY SOFTWARE OR OTHER SERVICES. BYTEPLANT SPECIFICALLY DISCLAIMS AND EXCLUDES ANY AND ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. DUE TO THE VARIETY OF USER EXPERTISE, HARDWARE AND SOFTWARE ENVIRONMENTS INTO WHICH CLEANMAIL MAY BE SUBJECTED, THERE IS NO WARRANTY FOR TECHNICALLY ACCURATE PERFORMANCE. THE USER ASSUMES ALL RISK OF USING CLEANMAIL. THE MAXIMUM LIABILITY OF BYTEPLANT WILL BE LIMITED EXCLUSIVELY TO THE PURCHASE PRICE.

LIMITATION OF LIABILITY NO LIABILITY FOR CONSEQUENTIAL DAMAGES. To the maximum extent permitted by applicable law, in no event shall

byteplant GmbH or its suppliers be liable for any damages whatsoever (including, without limitation, damages for loss of business profit, business interruption, loss of business information, or any other pecuniary loss) arising out of the use of, or inability to use, this software product, even if byteplant GmbH has been advised of the possibility of such damages.

LEGAL NOTICES CleanMail Home uses the spam filtering engine of the open source project SpamAssassinTM. 'SpamAssassin' and 'Powered by SpamAssassin' are trademarks of the Apache Software Foundation. The SpamAssassinTM open source project resides at http://spamassassin.apache.org. CleanMail uses the cairo (http://www.cairographics.org) and wxWidgets (http://www.wxwidgets.org) libraries.