Eridani Star System

MailStripper Pro 1.4

Release 1.4.1

Installation and Configuration Guide

Eridani Star System
Tel: **0131 208 0689**
Fax: **08701 600807**
Email: **mailstripper@eridani.co.uk**
WWW: **http://mailstripper.eridani.co.uk**

CONTENTS

WHAT IS MAILSTRIPPER?

MailStripper is an anti-spam package for Linux®, FreeBSD and Solaris®. The Linux binaries are statically built against XFree86-3.3.6 (but dynamically linked against glibc-2.1.3). The FreeBSD binaries are statically built on FreeBSD 5.3 and have been tested on FreeBSD 6.2. The Linux binaries will also run on NetBSD, with the Linux compatibility layer installed. The Solaris builds are built for SPARC (built on Solaris 10) and x86 (built on Solaris 9).

There are other anti-spam offerings available, but most of these either require specific versions of specific Mail Transport Agent (MTA) daemons or bolt into Procmail. Whilst these offerings may yield good results, they leave a lot of room for error and system downtime in their deployment, and can pose considerable risk to a currently operational system. MailStripper works differently, in that there are no specific requirements regarding your system's mail software. Indeed, MailStripper does not even require that your MTA is running on the same machine as itself. It could therefore be used to filter email coming into a Microsoft® Exchange server even though MailStripper is not a Windows product.

The only things MailStripper requires are the ability to listen on TCP port 25 and an MTA to talk to. This means that if an existing MTA is present on the same machine, it has to be configured to listen on an alternative port or network interface, with MailStripper configured to listen to the original address and port number, and to talk to your MTA on the new port number or interface address.

What MailStripper Is Not

MailStripper is not an anti-virus tool. It directly supports the use of either Frisk Software's F-Prot® package, Network Associates' McAfee® uvscan (either of which must be separately installed and licensed), or the open-source ClamAV package to scan incoming emails for viruses. This is a switchable feature, and MailStripper does not require this software to be present to perform as an anti-spam tool. In addition, the *avwrap* interface allows unsupported virus scanners and custom scanner configurations to be used.

MailStripper is not an MTA. It cannot do the work of a regular MTA package such as sendmail. It cannot replace your MTA. It listens on port 25, and acts as a semi-transparent filter, passing commands and data through to the MTA behind it. Your MTA is still responsible for sending the incoming email to the correct recipient. Depending on the email scan, MailStripper will either relay the reply from the MTA to the sender, or send its own reply instead (if, for example, the sender was blacklisted).

When configured to use F-Prot, MailStripper uses the Small Business Edition, as all that is required is the ability to scan a file. (The F-Prot for Mail Servers edition has its own tool for filtering mail, but is not a spam filter. MailStripper does not support the use of F-Prot for Mail Servers.) Personal users may download and use the Small Business Edition of F-Prot free of charge from Frisk Software's website. As a convenience, the features page of MailStripper's website includes links to their pages. MailStripper also supports the use of McAfee's "uvscan" and the open-source "clamav" antivirus packages.

Who Should Use MailStripper

The anticipated user base for MailStripper is the network system administrator. This document assumes a reasonable knowledge of email systems, IP addressing and network configuration. While MailStripper is not aimed at the home user as it cannot and will not run on Windows systems, those home users who use Linux, FreeBSD and Solaris are likely to have the required level of knowledge required to understand this document.

INSTALLATION

MailStripper is available for Linux, FreeBSD and Solaris. The instructions listed here are for the Linux version, but should apply equally to the FreeBSD and Solaris versions. NetBSD users can use the Linux binaries after installing the Linux compatibility layer. At a shell prompt and logged in as root, perform the following steps:

1. Change directory to /tmp:

   ```
   cd /tmp
   ```

2. Untar the tarball:

   ```
   gzip -dc /path/to/MailStripper-Pro_X.YY.tar.gz | tar -xvf -
   ```

3. Switch to the newly created directory:

   ```
   cd MailStripper-Pro_X.YY
   ```

4. Run the installation script:

   ```
   ./install.sh
   ```

5. Check to see if it works:

   ```
   mstripconf
   ```

   (X session users, use `xmstripconf`)

6. If it worked, then a menu will appear onscreen. `xmstripconf` users will get a control panel window.

At this stage, the binaries and initial files required by MailStripper should be installed on your machine. Feel free to explore the GUI noting that, until you enter your registration code using File->Register, no permanent changes to the settings can be made. Text-only users can explore the `mstripconf` interface; although changes will appear to be possible, they cannot be saved until you enter your registration code under the **`R`** option. Until a configuration file is saved, the daemon cannot operate.

An entry for MailStripper is added in `/etc/rc.d/init.d` or `/etc/init.d` on Linux and Solaris systems - this is to allow MailStripper to be started as part of your machine's start-up procedure. Also, if your distribution includes `chkconfig` then it is run to integrate MailStripper into the boot time initialisation sequence.

If you do not have X or are on a remote connection, try the text-mode `mstripconf` instead. The functionality is similar to `xmstripconf` – indeed, apart from the "tooltip"-style help messages there is nothing in `xmstripconf` that cannot be done in `mstripconf`, or vice versa.

REGISTRATION

Before MailStripper can be configured on your machine, it must be registered. The cost to register MailStripper on your machine depends on the type of licence you require.

Commercial licences cost **£79** (approx. US$177.00*) each, and this permits deployment for commercial use, whether it is for an office department or an organisation offering email accounts to the public whether for free or not. This permits use on a single server; if several machines are to be equipped with MailStripper, then a licence needs to be obtained for each. Note this is not a *per-CPU* licence, it is per machine, irrespective of the number of CPUs or mailboxes. This way your costs are completely under control. For mainframe servers which offer virtual servers (where each virtual server provides its own operating environment and for example a copy of Linux is running inside of each virtual server), then a licence is required for each virtual server requiring MailStripper's facilities since these virtual machines are considered separate servers.

Personal licences cost **£29.00** (approx. US$65.00*) each, and this permits deployment for personal use on your equipment at your home. This licence permits use on as many machines as required, providing they are all at your home location. This includes personal laptop computers unless the laptop is also used commercially elsewhere. This licence does not permit commercial use of any kind. Commercial use is defined here as using the computer to support any business activity. Home shopping and personal banking activities, for example, are not considered commercial use.

Evaluation licences are free, but are time limited to 28 days from when they are generated. When run on an evaluation licence, mstripconf and xmstripconf sessions will only run for two hours before they self-terminate. After the licence has expired, the system will revert to an unregistered state, and the daemon will pass email through as if globally whitelisted. This means that MailStripper will stop filtering out spam or viruses, but will not stop your mail system from receiving email. Evaluation licences are available upon request from the MailStripper website, and your evaluation licence will be automatically emailed to you. Only one evaluation licence may be obtained per customer in any six month period, and both the website and the software will keep track of this.

These registration fees are a one-off payment for the software, as MailStripper is *not* a “rent-ware” product. The only time-limited licence codes are the evaluation licences, available free of charge. That said, the blocklist update facility is a subscription service, to which your purchase will give you a 6-month subscription. If the subscription is not taken up, the software will revert to pulling the “free” update list which is updated monthly.

Paid licences may be obtained online at **http://www.eridani.co.uk/MailStripper/** and the licence codes will be emailed to you. If you would prefer to pay by cheque (drawn on a UK bank), bank transfer or UK postal order, please contact us for details – in this case the licence codes will be emailed when payment clears.

You can also purchase licences over the telephone via **Reg.Net** with most major credit cards. US customers call **1-800-999-2734** (toll-free). Other customers please call **+1-719-576-0123**. (At the time of writing, UK users can call this at local rate by dialling through 0845 2 441 441 for no further charge.) You will typically receive your licence code by email within 2 working days. The product codes used by Reg.Net are 13031 (commercial licence) and 13038 (personal licence). Please note that Reg.Net may add sales tax or VAT to the above prices depending on your location, and they charge in US dollars.

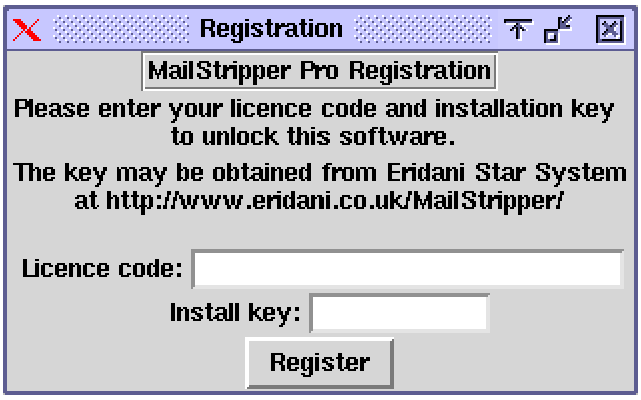
If you prefer, we also accept PayPal for online payment. Again, you will receive your licence by email typically within 2 working days.

When your licence codes arrive in your email, **KEEP THEM SAFE**. A hardware change to your machine or a kernel upgrade may easily result in MailStripper no longer recognising the stored key, and will require you to re-enter your registration code. The codes emailed to you will be valid for this.

Your key is valid for all MailStripper releases and upgrades up to and including version 1.X, including any minor version release. For example, users of versions 1.0.x to 1.3.x may upgrade to 1.4.1 free of charge – the software will automatically see and recognise the licence code. Version 2.0 and later may or may not be licensed using the same codes but, if the licensing scheme changes, a discounted upgrade path will be made available for existing users.

* Reg.Net now charge California sales tax, this is included in the US$ prices shown.

If using X-Windows: Once you receive your licence codes in your email, run the xmstripconf GUI and click on File -> Register. A dialogue box like the one below will appear, inviting you to enter your licence code (based on your email address) and the installation key which is tied to your licence code. If these details are entered correctly, clicking on **Register** will show a pop-up box informing you that your copy of MailStripper has now been registered successfully, and the Registration window will close. If you made a mistake, the Registration window will not close, and a pop-up box will appear notifying you your credentials were not valid. Click on **OK** and try again.



Registration

MailStripper Pro Registration

Please enter your licence code and installation key to unlock this software.

The key may be obtained from Eridani Star System at http://www.eridani.co.uk/MailStripper/

Licence code:

Install key:

Figure 1: Registration window

The licence code field contains your email address and suffix (e.g. `/P`, `/C02`), which identifies the type of licence. The installation key is the code generated using the licence code and is unique to that licence code. Both are included in the email sent to you. Once entered, click on **Register**.

The File->Register menu will not be displayed if MailStripper is running with a paid licence. When running on an evaluation licence, the menu option is still present, to allow a seamless upgrade to a full licence. In addition, MailStripper will not even need to be restarted.

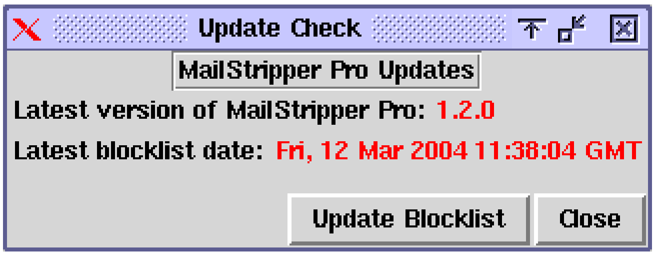
If using the text-mode mstripconf tool: Press the `R` key and enter the licence details at the prompts.

When licensed: The `R` key in mstripconf will show the current software licence type. If an evaluation licence is in use, it will also offer the opportunity to upgrade to a full licence.

If the software complains of invalid credentials and you are trying to use an evaluation code, this indicates that an evaluation licence has been entered into it within the past six months. If you require an extension to your evaluation period, contact us and we may be able to issue you with an evaluation code that will work. This will not affect paid licences in any way.

UPDATES

MailStripper is constantly under development to improve its ability to detect spam. To this end, the Help -> Latest Version will display a dialogue box which queries the MailStripper website and displays a summary of the latest updates, like this:



Latest version of MailStripper Pro: 1.2.0

Latest blocklist date: Fri, 12 Mar 2004 11:38:04 GMT

Figure 2: Update Check dialogue box

The **Update Blocklist** button downloads the latest blocklist file from your chosen location (see page 10), replacing your current blocklist. The previous blocklist is saved with a timestamp (inside `/usr/share/mailstripper`), allowing you to revert if you choose. In mstripconf, this functionality is available using option **U** from the main menu, pressing **D** to download the new blocklist.

See also the "**Blocklist Auto-Update**" feature described below.

Software Upgrades

From time to time, newer versions of the MailStripper software will be released (together with new versions of this document if required). MailStripper will update your configuration file during the upgrade, automatically ensuring that any new configuration options are present whilst leaving the existing settings unchanged. All other configuration files remain untouched, as the default ones are only installed if not already present.

If you are upgrading from a 1.0.x or 1.1.x release: Your old blocklist file will be renamed blocklist.11X.orig in MailStripper's internal format, and a clear-text copy will be made in the `/etc/mailstripper/blocklist.11X` file. Indeed, the old configuration directory `/usr/share/mailstripper` has been deleted and its contents moved in their entirety to `/etc/mailstripper` as many sysadmins keep `/usr/share` as a read-only partition.

Unlike the upgrade to 1.2.0, if upgrading from a version older than 1.3.0 the 1.4.0 upgrade will automatically adjust your blocklist update URL to use the "Use Eridani blocklist" option, which calls a CGI to determine whether to send you the subscription or free blocklist. If you used a non-standard URL this is preserved and is available for use by choosing "Use Blocklist URL". The 1.2 file at http://www.eridani.co.uk/MailStripper/blocklist12 will be maintained as a copy of the free blocklist.

ADDITIONAL TEXT-MODE FUNCTIONALITY

In addition to providing a text-mode alternative to the GUI configuration tool, mstripconf also provides additional command-line functionality that can be used within scripts.

Configuration Management

The text-mode mstripconf tool has options that allows you to dump and adjust the configuration file used by MailStripper.

Syntax: **`mstripconf -showconf`**
Syntax: **`mstripconf -set <variable> <value>`**

The variable and value are both case specific, and all entries shown in "mstripconf –showconf" apart from EVALUSED can be changed. Unrecognised variables are silently ignored.

Sender Whitelist Update

The mstripconf tool also allows the sender whitelist to be updated from the command line or a script.

Syntax: **`mstripconf -whitelist <email address>`**

The supplied email address must be the sender's email address as displayed in the "`Received: by mailstripper-deliver`" header line, and not that in the normally-displayed "`From`" header line.

IP Blacklist management

The tool additionally allows entries to be added to or removed from the IP blacklist, either from the command line or in a script. The remove feature is intended primarily for use by housekeeping scripts that check IP addresses present in the file against a DNS blacklist, to remove those added to the DNS blacklist since they were caught in the honeypot.

Syntax: **`mstripconf -ipblock <IP address>`**
Syntax: **`mstripconf -ipunblock <IP address>`**

In either case, the IP address may contain a trailing "." – if no trailing dot is supplied, it will be added to ensure that the matching behaves as intended. When removing an IP address, it will be matched whether or not a trailing dot is used.

An example script showing how these options can be used will be found on our FTP site, at `ftp://ftp.eridani.co.uk/pub/MailStripper/contrib/cleanipblock.sh` - this script is in the public domain, so you are free to use it how you wish.

Blocklist Auto-Update

Use **cron** as **root** to run either "**`mstripconf -update`**" or "**`mstripconf -nupdate`**" at your chosen polling times. The `-nupdate` option is identical to the `-update` option in all respects other than that `-nupdate` does not make a timestamped backup copy of your old blocklist file.

For example:

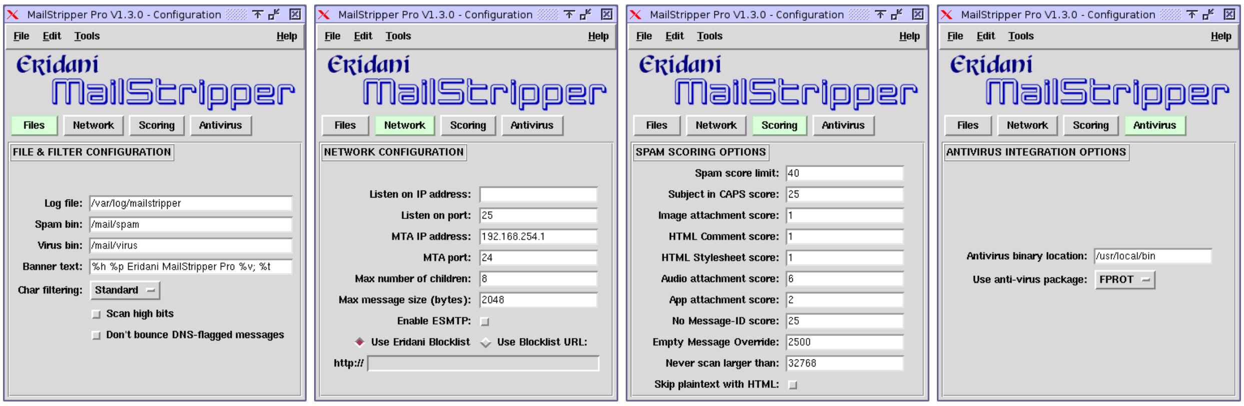
```
27 5 * * * /usr/sbin/mstripconf -nupdate
```

This will update the blocklist at 5:27am without making a backup of the previous blocklist file.

If you are using the (default) download location at Eridani, be aware that this is updated at 3:30am (UK time) if any changes are made during the preceding day.

CONFIGURATION

MailStripper is a fairly complex system, and as such has a number of configuration options that affect its behaviour.


MailStripper Pro V1.3.0 - Configuration
File Edit Tools Help
Eridani MailStripper
Files Network Scoring Antivirus

FILE & FILTER CONFIGURATION

Log file: /var/log/mailstripper
Spam bin: /mail/spam
Virus bin: /mail/virus
Banner text: %h %p Eridani MailStripper Pro %v; %t
Char filtering: Standard
□ Scan high bits
□ Don't bounce DNS-flagged messages

NETWORK CONFIGURATION

Listen on IP address:
Listen on port: 25
MTA IP address: 192.168.254.1
MTA port: 24
Max number of children: 8
Max message size (bytes): 2048
Enable ESMTP: □
● Use Eridani Blocklist ○ Use Blocklist URL:
http://

SPAM SCORING OPTIONS

Spam score limit: 40
Subject in CAPS score: 25
Image attachment score: 1
HTML Comment score: 1
HTML Stylesheet score: 1
Audio attachment score: 6
App attachment score: 2
No Message-ID score: 25
Empty Message Override: 2500
Never scan larger than: 32768
Skip plaintext with HTML: □

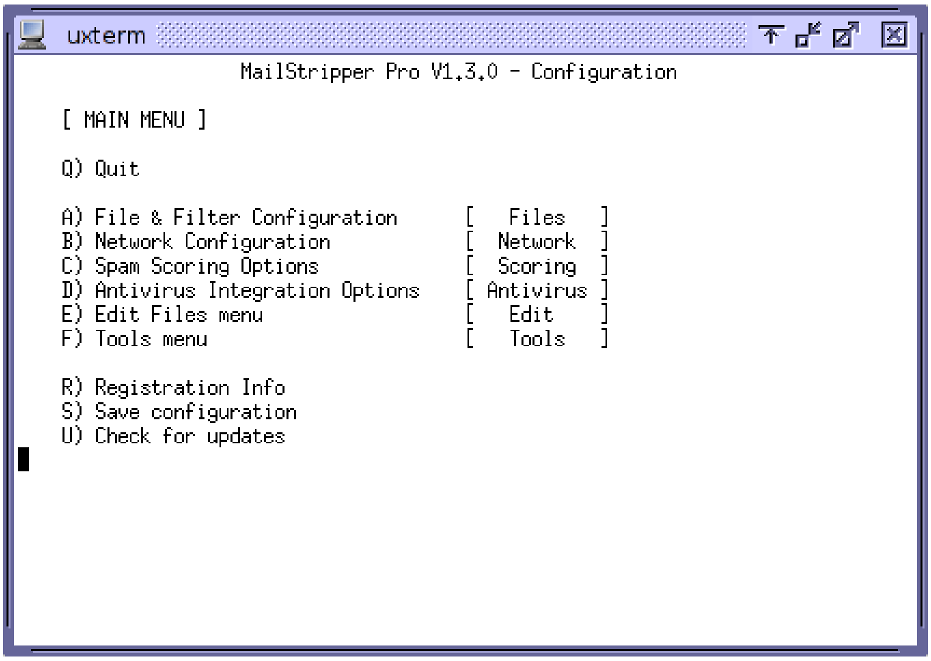
ANTIVIRUS INTEGRATION OPTIONS

Antivirus binary location: /usr/local/bin
Use anti-virus package: FPROT

Figure 3: xmstripconf main control windows

The above windows might appear a bit cryptic at a first glance. However, the application supports pop-up context-sensitive help (similar to "tooltip" help common in Windows). For those who do not like this help mechanism, this can be turned off under the Help menu option, and this setting is saved in the configuration file. Each of these settings is explained in this document.

The mstripconf and xmstripconf tools require root access to modify the configuration files. If a user other than root runs them, they will show an error message before exiting.



```
                    MailStripper Pro V1.3.0 - Configuration

   [ MAIN MENU ]

   Q) Quit

   A) File & Filter Configuration      [   Files   ]
   B) Network Configuration            [  Network  ]
   C) Spam Scoring Options             [  Scoring  ]
   D) Antivirus Integration Options    [ Antivirus ]
   E) Edit Files menu                  [   Edit    ]
   F) Tools menu                       [   Tools   ]

   R) Registration Info
   S) Save configuration
   U) Check for updates
```

Figure 4: mstripconf (text mode) main menu

The text-mode configuration tool uses a layered menu to replicate the organisation of options in the GUI and make it easier to navigate.

File Menu

The options under the File menu are:

Register

This opens the registration dialogue window, and is hidden when MailStripper has a valid licence installed.

Save Config

Saves the current configuration. Greyed out if MailStripper has no valid licence installed. While the configuration file is being saved, the menu bar will change colour to red.

Exit

Exits the configuration tool.

This menu may be "torn off" into its own small window.

Edit Menu

The options under the Edit menu are:

Blocked Words List - `/etc/mailstripper/blocklist`

This file contains the word list used by the spam scanner, and their associated bias. Negative values are valid; for example use –2 for a reduction of two points.

Email Blacklist - `/etc/mailstripper/senders.deny`

Email addresses (or parts of addresses, including domains) from which emails are rejected. The rejection is handled as if the recipient address did not exist on your system. This address must be given as seen by the `MAIL FROM` SMTP command – this will be recorded in the log file.

Email Whitelist - `/etc/mailstripper/senders.allow`

Email addresses (or parts of addresses, including domains) from which emails are always accepted, unless they are found to contain a virus. This address must be given as seen by the `MAIL FROM` SMTP command – this will be recorded in the log file.

Recipient Whitelist - `/etc/mailstripper/recipients.allow`

Email addresses (or parts of addresses, including domains) to which emails are always accepted, unless they are found to contain a virus. Only useful for local users (or ones which you are a relay for) who want an unfiltered email feed. This address must be as seen by the `RCPT TO` SMTP command – this will be recorded in the log file.

Recipient Taglist - `/etc/mailstripper/recipients.tag`

Email addresses (or parts of addresses, including domains) to which emails are subject-line tagged (see `#TAG`) if found to be spam, but quarantined if they are found to contain a virus. Only useful for local users (or ones which you are a relay for) who want a tagged email feed (so they can quarantine locally). This address must be as seen by the `RCPT TO` SMTP command – this will be recorded in the log file. Don't use this in conjunction with `#TAG`.

IP Blacklist - `/etc/mailstripper/badips`

Connections from these IP addresses (or netblocks) will be handled as if the recipient user did not exist on your system. These IPs should be suffixed with a . character to prevent partial matching.
Additionally, a DNS blacklist may be used here by prefixing their suffix with "`DNS:`", for example include a line such as:

```
DNS:sbl-xbl.spamhaus.org.
```

Any response from the blacklist in the form of an IP address (irrespective of what the address returned is) is treated as a blacklist match.

Honeypot List - `/etc/mailstripper/honeypot`

This file contains two separate item types. These are the honeypot email addresses, and IP addresses (or netblocks) which are never to be blacklisted. The IP addresses must be prefixed with a `!` character, and like the IP blacklist suffixed with a . to prevent partial matching. The email addresses should be listed, and like the recipient whitelist they must be entered as seen by the `RCPT TO` SMTP command – this will be recorded in the log file.

Permitted Domains List - `/etc/mailstripper/domains.allow`

Domains that you accept mail for should be listed in this file. Its purpose is to prevent unauthorised relaying through mailservers that for one reason or another insist on permitting relaying through a local address despite being configured otherwise.

All these menu options bring up editor windows with their respective control files. After making any changes, be sure to save your changes using the editor window's own File->Save option. While the file is being written, the menu bar will change colour to red.

The above files which accept email addresses may have the entries prefixed or suffixed with a `!` character. Internally, the address the file entries is matched against will have both (e.g. `!wibble@blah.com!`). Ordinarily, if the file entry matches any part of the address then it will be considered a match. Using `!` forces the match to be with an edge of the string – or by using both, an exact match.

This menu may be "torn off" into its own small window.

Tools Menu

The options under the Tools menu are:

Manage Plug-Ins

This opens a new window where various plug-ins to MailStripper may be controlled. Please see the section on Plug-Ins for further information on this feature.

Help Menu (xmstripconf only)

The options under the Help menu are:

About

Displays a licence-specific box detailing what the program is and how it can be used. It also displays whom the package is licensed to and what sort of licence has been registered. This information can be obtained via "Registration Info" in the text-mode mstripconf tool.

Latest Version

This brings up a pop-up window which displays the current version of MailStripper as held on the web server, and the latest version of the blocklist file. For an example, see Figure 2. The blocklist may be updated from the server using this window. This is available through "Check for updates" in mstripconf.

Use Tooltips

This toggles whether or not you want "tooltip"-style context-sensitive help to be used on the main panel. Some people find them useful, some find them very annoying – so this setting is entirely user switchable and saved with the configuration. These don't exist as pop-ups in mstripconf, but each sub-menu page gives instructions on how to bring up context-sensitive help on the various options within that menu.

Main Configuration Panel

The options available are:

Log file (Files tab)
This is the location of the MailStripper log file.

Spam bin (Files tab)
Messages caught as spam get placed in this file. It is in mbox format, such that mail applications like Pine can handle directly. It's worth reviewing this mail folder periodically in case MailStripper has accidentally caught legitimate email. Alternatively, if this is specified as an email address (which must be local to your site, or to which your site is a configured relay), spam mail will be sent there instead of the intended recipient. For example, `spambin@example.com` would be a suitable name for a role account for handling trapped spam. Of course, be careful on your choice of email clients used to handle this account as many spam messages use "web bugs" in the HTML body of the message.

MailStripper also supports per-user and per-domain spam bins. This is selected by using `%u` (user name) and `%d` (domain name) to substitute the relevant part of the recipient's email address. This works both for the mbox-type spam bin and the spam-handler email address. For example, using **`/var/spool/spam/%d/%u`** will put spam addressed to `foo@example.com` in `/var/spool/spam/example.com/foo`, and spam addressed to `wibble@blah.co.uk` in `/var/spool/spam/blah.co.uk/wibble`. The substitution "`%t`" is also supported, to provide a unique time and PID stamp. This is only useful as part of the leaf name, not the directory path.

Use of the "**`#TAG`**" setting here is deprecated, but still functional. For a more flexible option to perform the same thing, see the Recipient Taglist file.

Virus bin (Files tab)
Messages found to contain a virus are placed in this quarantine file. Normally there should be no need to read this mail folder, and if you are happy to, this can be set to `/dev/null` to effectively delete all virus-infected mail.

Banner text (Files tab)
This allows a custom SMTP welcome banner to be sent to the connecting client. The default is "`%h %p Eridani MailStripper Pro %v, %t`". The 220 success code is prefixed to this. The expansions are:
`%h` Hostname
`%p` Protocol (SMTP or ESMTP)
`%v` Version of MailStripper
`%t` Current time and date.
NOTE: In order to be compliant with RFC821 (SMTP) and its successors, **`%h`** must be the first item specified in your banner string.

Char filtering (Files tab)
This determines how characters are filtered out before scanning. The options are **Standard**, **Std+Space** and **NonAlphaNumeric**. When Standard is chosen, the list of characters shown on Page 16 are filtered out of the message prior to scanning. Std+Space filters these characters and space characters out. NonAlphaNumeric filters out all characters apart from letters, numbers and newlines.

Scan high bits (Files tab)
This performs a quick check on the number of high-bit characters are present in the message, and is triggered by a lot of Far Eastern spam. This test is performed after folding the accented / look-alike characters to their unaccented equivalent. **DO NOT** enable this if you expect to receive legitimate mail in Unicode/Far Eastern character sets.

Don't bounce DNS-flagged messages (Files tab)

Normally, when a sender is listed in a DNS blacklist (see the IP Blacklist file above), the messages are rejected with a "`550 No such user`" error message. With this option selected, MailStripper will quarantine such messages instead, as if it had simply failed the scan test. Note the scan test is not performed, as the DNS blacklist implies the message is extremely likely to be spam.

Listen on IP address (Network tab) [*Restart required*]

If set, MailStripper will only listen on the network interface owning this IP address. Otherwise, it will listen on all IP interfaces present.

Listen on port (Network tab) [*Restart required*]

MailStripper will listen to this TCP port for incoming SMTP connections. For almost all applications this should be left on **25**.

MTA IP address (Network tab)

The IP address of your "real" MTA (e.g. Sendmail, MS Exchange). Do not use 127.0.0.1 (localhost) or any local IP address that your MTA is configured to see as local, as your mail server must see this as a remote address. It is recommended that users who run their MTA on the same machine as MailStripper configure the *dummy* interface or an Ethernet alias (e.g. eth0:1) with an alternate IP address in the RFC1918 range which will not be recognised as local (and not listed as allowing relaying) by your MTA.

If you use a POP3 mail store in conjunction with fetchmail to download your messages, using 127.0.0.1 will work provided Port 25 is firewalled from **any** outside connections.

MTA port (Network tab)

The TCP port that your MTA is listening to for incoming SMTP connections. If MailStripper is running on the same machine as your MTA this needs to be something other than 25, and your MTA needs to be configured to match. Sendmail users may get some mileage from the `updateSendmail.sh` script supplied in the distribution package.

Max number of children (Network tab) [*Restart required*]

This sets the maximum number of scanning processes that may run at any one time. The more scanning processes running, the less CPU time is available to each, causing them to slow down. If the number of running scanning processes reaches this value, new connections are rejected with "`421 Server Too Busy`". This is a temporary error code telling the sender to try a secondary mail server if one is listed in the DNS, or to queue the email message for retrying later. If the source IP is blacklisted, this value is effectively halved for that connection, lowering the priority below that of non-blacklisted connections.

Spammers apparently tend not to retry failed server connections as this would slow down their attempts to send their junk – but genuine mail senders will always retry sending as per the SMTP specification.

Max message size (Network tab)

This sets the maximum size of message we're prepared to accept. This is only effective in ESMTP mode and when the sender declares the size. Therefore this should not be relied upon to prevent over-size mail coming in. If the message doesn't declare its size, it will be accepted irrespective of its length (resources permitting) and will not be rejected after being received.

The idea being this is to reduce the load on bandwidth, and if we've received the whole thing we might as well deliver it. If the maximum size must be enforced (as required by our sister operation `http://www.MailMeAnywhere.com/`) then this can be achieved using a custom plug-in.

Enable ESMTP (Network tab)

With this enabled, a subset of ESMTP (Enhanced SMTP) commands will work. The default is ESMTP disabled.

Network Timeout (Network tab)

This sets the timeout after which the connection will be dropped when no data is received. The default is 300 seconds (5 minutes).

Update Blocklist From (Network tab)

This toggles between choosing the Eridani server, or another server by URL.
This menu item is only present in this form in the text-mode `mstripconf` tool. The GUI has two radio-button options instead, **Use Eridani Blocklist** and **Use Blocklist URL**.

Blocklist Update URL (Network tab)

This option is only accessible when the Update Blocklist From setting is set to "URL". This specifies the location to which MailStripper looks when asked to update its blocklist, whether performed through the `U` option in mstripconf, the Help->Latest menu in xmstripconf or the auto-update facility. The value "default" is no longer supported, instead set Update Blocklist From to Eridani.

Spam score limit (Scoring tab)

The threshold at which a message's score determines it as being spam.

Subject in CAPS score (Scoring tab)

The extra score given to a message if its subject line is entirely in capitals. This is added after the score is normalised.

Image attachment score (Scoring tab)

The additional score given to a message when an image attachment is found. This is added before normalisation.

HTML comment score (Scoring tab)

This score value is added to the total before normalisation each time an HTML comment (or other HTML junk) is found.

HTML Stylesheet score (Scoring tab)

The extra score added before normalisation when an HTML stylesheet is found inlined in the email. A lot of HTML spam messages include these.

Audio attachment score (Scoring tab)

The additional score added before normalisation when an audio attachment is found. Most email attachments claiming to be audio are in fact viruses.

App attachment score (Scoring tab)

The extra score added before normalisation when a miscellaneous binary attachment is found.

No Message-ID score (Scoring tab)

This score is added after normalisation, if the scanned message did not include a `Message-ID:` header.

Empty Message Override (Scoring tab)

If the email message body is completely empty, set the score to be this value instead of running the scanner.

Never scan larger than (Scoring tab)

Large message bodies can take a long time to scan on older machines. Inbound messages which are larger than this (after removing non-text attachments) are not scanned. Plug-ins are still called to examine the message irrespective of this.

Skip plaintext with HTML (Scoring tab)

If enabled, and an email contains both plain-text and HTML components, the plain-text is not scanned, only the HTML. The default is NO, which causes both parts to be scanned.

Antivirus binary location (Antivirus tab)

The location of your antivirus binary. This may be a symlink to another location. For example here at Eridani where we use ClamAV it lives in `/usr/local/bin`.

Use antivirus package (Antivirus tab)

This drop-down determines whether or not MailStripper should use a virus scanner, and if so, which one. Please note that MailStripper does not include licences for any antivirus packages, and users will need to purchase licences for their chosen package separately. The open-source ClamAV package is supported by MailStripper and is available from `http://www.clamav.net/` - the clamd daemon mode of operation is by far the fastest way of running this.

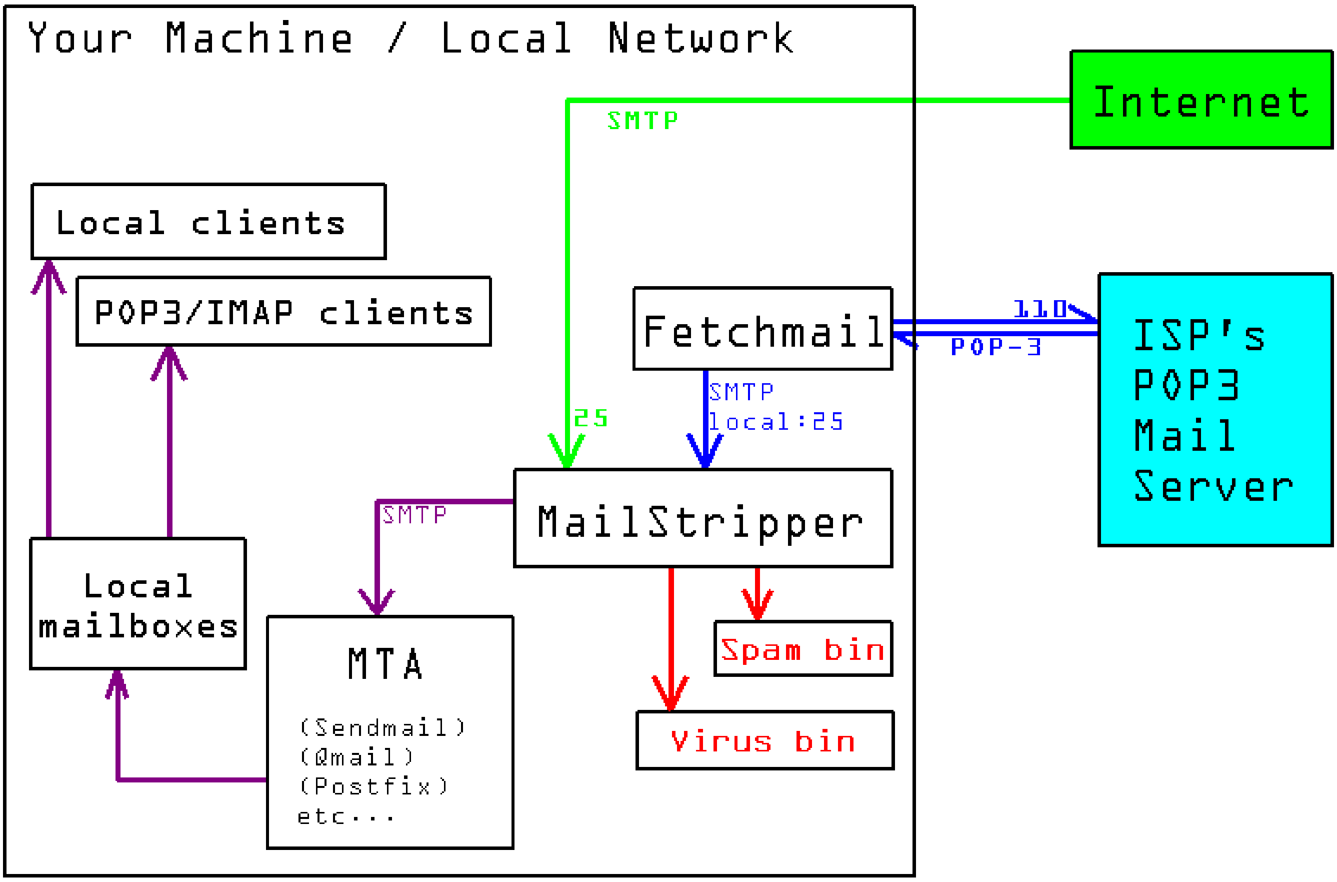
Preferred text editor (mstripconf text-mode tool only, under Edit Files menu)

This defaults to the location of the `vi` editor on your system. If you prefer to use a different editor (e.g. emacs, pico) specify the full path to it here. Although this option is only visible (and meaningful) within the text-mode tool, the setting is preserved when the GUI tool (xmstripconf) is used.

The tag **[*Restart required*]** indicates that MailStripper needs to be restarted for changes in these options to take effect. If this tag is not shown, the setting change will take effect immediately except for message scans currently in progress.

NETWORK CONFIGURATION OVERVIEW

This diagram outlines the way MailStripper integrates into various mail system options:



When using Fetchmail to retrieve mail from a POP3 mailstore, be sure to set up the associations between your ISP mailbox (e.g. *john.smith@example.com*) and your local username (e.g. *jsmith*). Of course, if you currently use fetchmail to retrieve your mail and inject it into your SMTP daemon you will not have to reconfigure this – after installing MailStripper, fetchmail will then find itself talking to MailStripper instead of your real MTA.

Not all the options above need to be in place on all systems. For example, sites with an SMTP-only feed need not implement the fetchmail path, and sites with only a POP3 mail feed need not implement a net-to-MailStripper SMTP path (but in this instance we recommend that port 25 is then blocked to anything except localhost using some form of firewall).

If your mail server has only one IP address (excluding localhost), you will need to reconfigure your MTA to listen to an alternative port – we've included a script "updateSendmail.sh" that does this for Sendmail setups. Port 24 has been assigned by IANA for local mail services, therefore use of this port should minimise port clashes later on. This will also require that your email clients be reconfigured to send their outgoing email to the new port.

An alternative configuration may be used if your mail server has more than one physical network port – e.g. static IP Ethernet or PPP connection to the Internet, and a separate Ethernet connection to the LAN.

If this is the case, configure MailStripper to listen specifically on your public IP address on port 25, and configure it to use the mail server on your local IP address port 25. In addition, configure the mail server to only listen on <local IP>:25 and localhost:25. It is perfectly OK to use RFC1918 addresses within your localnet for this.

This way, outgoing email from your LAN will go directly to the mail server for onward mailing, and incoming mail will go via MailStripper, without the need for your internal mail clients to be reconfigured to operate on a separate port.

IMPORTANT: MailStripper is not intended to handle outgoing email. Due to its proxy nature, your MTA will see the source address as your MailStripper IP, not the true sender address and therefore cannot tell what is local and what is not. Attempting to route all outbound email through the same MailStripper set-up that is also handling all inbound email will result in your mail server becoming an open relay. If you only have one IP address, and your internal clients use your public address to send outgoing email, it is far better to either reconfigure the clients to use an alternative port number (e.g. 24) for outbound SMTP, or run MailStripper on a separate machine to your email server. This will also result in a reconfigure, either of your clients or (if you use inbound NAT with port-forwarding, e.g. on an ADSL line) your router to forward inbound SMTP to the new machine running MailStripper. A well-configured MailStripper set-up will cause an email client trying to send outbound email through it to fail with a Relaying Denied error.

However, if your SMTP server supports it, some level of support is in place for SMTP AUTH tunnelling through MailStripper, and if you only have the one public-facing IP address, may allow you to use your server for remote users using SMTP authentication to send email.

THE SCORING SYSTEM

The scoring system in MailStripper Pro is the core of how it determines what is spam, and what is not. It is based upon a keyword search, with some modifications and extra rules in place.

The scores are read in from the blocklist file (which may be tailored to suit your requirements using the GUI interface), along with their individual weightings. The headers are not searched with the exception of the subject line.

Incoming messages which are encoded in Quoted-Printable, Base64 (MIME) or HTML escapes (such as `&#65;` or `&#x2A;`) will be decoded before processing. Accented characters are mapped to their non-accented equivalent.

The rules are basically:

If the word is found on the subject line, 3x(score) is added to the total.

If the word is found in UPPERCASE, 3x(score) is added to the total. If this word in uppercase is found on the subject line, 9x(score) is added to the total.

If various attachment types are found, their appropriate score is added to the total, and the number of lines the attachment occupies is not added to the total (except those of type text/*). See the previous section for details on this.

A score of 0.2 is also added for every word that appears without a vowel in it, to assist catching the strange spams that arrive with no actual readable text in them. This score is low since there are situations where it would otherwise be tripped by genuine content.

The score is then normalised against the message length, in number of lines. Binary attachments are not counted towards the length, nor are blank lines in the message. A bias of 120% is added if the main body of the message is BASE64 encoded. The header is scanned, but does not count towards the counted length of the message.

If the message body is blank, the header is considered to add a length of 1, and the before-normalisation score is set to 5.

After normalisation, some more adjustments are made, if the message has a subject line entirely in uppercase, or does not contain a Message-ID header.

Some characters are filtered out of the message before the scanning process is run. These characters are:
`` ` ^ # ~ * _ ’ : ; , . - `` (note: this list has changed since 1.2.4)
If the **Char filtering** option on the Files tab is set to “Standard”, then only these characters are removed. If this is set to “Std+Space”, all spaces in addition to the above characters are removed. Setting this to “NonAlphaNumeric” removes ALL characters apart from letters, numbers and newlines.

A word of warning: Filtering out spaces may help catch words where the spammer has attempted to hide them from filters by adding spaces. However, it may also cause erroneous matches where part of one word and part of another word join together to form a string which contains a word present in the blocklist file.

The words in the blocklist file may use Perl-style regular expressions. For example,

```
1 rec(ei|ie)v(e|ed|er|ing)
```

will attract a score of 1 for each occurrence of “receive”, “received”, “receiver” and “receiving”. It will also catch misspelled versions using “recieve” instead of “receive”.

If you want to catch “receive” but not “received”, use something like

```
1 receive[^d]
```

Another that is in our blocklist file is one that catches a lot of the illegal pharmacy spam:

```
10 v[ail1][a@i][gq]r[a@]
```

New to version 1.3.0, prefixing the score value with the = character will skip the upper-case check on that rule. This is also true for the Local Blocklist plug-in (see later).

BLACKLIST AND WHITELIST PRIORITIES

The various mail filters have a pre-determined order of priority. In decreasing order of precedence, they are:

1) **The Permitted Domains list**
This works by examining the envelope recipient address (the SMTP "`RCPT TO:`" bit). It aims to prevent relaying when using difficult mail servers which seem to insist on allowing local addresses to relay mail irrespective of its configuration. Recipients not matching any entry in this file are rejected with a "`550 5.7.1 Not local`" error. Using a "`@`" in the file on a line of its own will match everything. Note that this file is not a whitelist; rather, it is an inverse blacklist. That is, if the recipient is *not* on the list, the message is refused.

2) **Anti-virus**
If the message contains a virus, it is blocked. The virus can be quarantined, but MailStripper will send a "`550 5.1.1 Rejected: <reason>`" back to the sending mail server. (MailStripper will never create a bounce message itself.)

3) **Sender Whitelist**
Be aware this also uses the envelope details (the SMTP "`MAIL FROM:`" bit) and not the displayed "From" header (as this hasn't yet been seen, and is slightly harder to forge).

4) **Recipient Whitelist**
Again, this uses the envelope recipient address (the SMTP "`RCPT TO:`" bit), not the displayed "To" header.

5) **IP Blacklist, including DNS Blacklist**
This does not examine the header, as these are easily forged. Instead it uses the socket peer address (the source address reported by the host UNIX operating system). This is extremely difficult to forge.

6) **Sender Blacklist**
As with the sender whitelist, this looks at the SMTP "`MAIL FROM:`" bit.

7) **Honeypot list**
Like the recipient whitelist, this uses the SMTP "`RCPT TO:`" bit, and populates the IP Blacklist with sender IP addresses. (The file also allows permitted relays to be specified, to ensure they are never blacklisted.)

8) **Plug-in interface**
The plug-ins are run in alphabetical order. Any that return a whitelist code will stop further plug-in processing, and will cause the content filter to be bypassed. Any that return a blacklist code will stop the content filter from running, but will not stop further plug-ins (as a later one may return a whitelist code). When one plug-in returns a whitelist code and another returns a blacklist code, the whitelist takes precedence.

9) **Content Filter**
Lastly, the message is scanned by the content filter, and is quarantined or permitted depending on the results of the scan.

SCORE ANALYSIS TOOL

This tool is based around the spam search core used by MailStripper. This tool, however, when run on a file, shows the resulting breakdown and the score as it is accumulated. Use this tool on saved-out spam messages that were missed, or genuine messages that were picked up by the spam scanner, to enable you to fine-tune the scanner or amend your local blocklist file.

Note that a very small score difference between what was written in the headers and what score shows may occur, especially if there are any trailing blank lines or the mail server applied any translation on the body of the message.

Syntax: **`score <filename> [<filename> […]]`**

THE AVWRAP INTERFACE

The *avwrap* interface is a gateway for enabling the use of unsupported virus scanners.

All that is required of it is that it returns no output if the message is deemed to be clean, or it returns one line if the message is to be blocked and quarantined.

The syntax of the return line is:
`Infection: <reason>`

e.g. **`Infection: Win32/MyDoom.A@mm detected.`**

The script should exit with reason code 0 in all cases.

The script may be written in any language (it may even be a true binary), but must be named **`avwrap`**. However, for most purposes a shell script, or Perl or Tcl scripts will be more than adequate.

An example script is located in `/usr/local/bin/avwrap`. Implemented as a Bourne shell script this blocks Win32 binaries and scripts, matching by the filename extension. MIME types are ignored as most viruses lie about the MIME type.

This example script is in the public domain; you may use it and modify it as you wish.

In order to use the avwrap interface, the AVWRAP option must be chosen in the anti-virus package drop-down box or option T in the text-mode interface, and the appropriate location set in the antivirus location path (option P). If the script cannot be found or does not have execution permission, a warning will appear in the MailStripper log file and the message will be handled as if no AV scanner was configured.

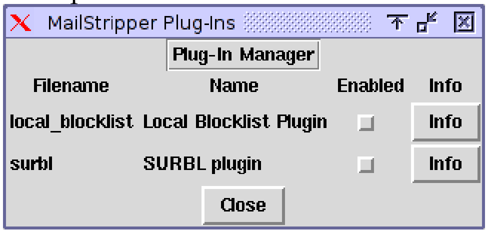
This interface is also good for using custom configurations, such as a “brute force” script that attempts to decrypt a password-protected zip file used by some more recent viruses. F-Prot will ordinarily recognise a suspicious file in a password-protected Zip file but is currently unable to scan it without assistance.

THE PLUG-IN INTERFACE

MailStripper now supports the use of plug-ins to add new scanning mechanisms to the spam filter.

Two such plug-ins are provided as standard as part of the distribution, **SURBL** and **local_blocklist** and are installed in the plug-ins area /etc/mailstripper/plugins. Another two, badwords and linked_images are demonstrations of the API and are in the SamplePlugins directory of the package tarball, but are not installed as standard. These last two are in the public domain; you may use them and modify them how you wish.

Clicking on Tools -> Manage Plug-Ins will open a new window like:

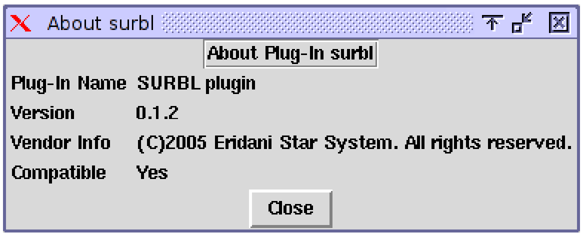


MailStripper Plug-Ins

Plug-In Manager

Filename	Name	Enabled	Info
local_blocklist	Local Blocklist Plugin	□	Info
surbl	SURBL plugin	□	Info

… and clicking on Info will open a dialogue like:



About surbl

About Plug-In surbl

Plug-In Name	SURBL plugin
Version	0.1.2
Vendor Info	(C)2005 Eridani Star System. All rights reserved.
Compatible	Yes

The "Enabled" selection box must be selected for each plugin that is to run during the scan phase of a message. Changes take effect when the config file is re-saved; there is no need to restart MailStripper.

SURBL

The SURBL plugin implements the Spam URI Realtime Blocklist at http://www.surbl.org by looking at the URIs contained within the body of messages.

This plug-in reads an optional config file at /etc/mailstripper/surbl.config, which if not found is assumed to contain the following entries:

```
set blname multi.surbl.org
set scinc 10
set once 1
set refuse 0
set honeypot 0
```

Also, when a config file is present but does not contain all the above entries, those which are missing are assumed to contain the entries above.

The entries are defined as:

blname: The name of the SURBL blocklist to use.
scinc: The score increment for each spamvertised URI. Only effective if refuse = 0.
once: Boolean, to set whether spamvertised domains are only counted once per message, or once per mention. Default: 1 (once per message) Setting this to 0 may have a considerable performance hit.
refuse: Boolean, if 1 causes any match to return a 2000 (refuse) code. Default: 0
honeypot: Boolean, if 1 (and refuse = 1) causes any match to return a 3000 (refuse and block) code. Default: 0.

If the IP address happens to be listed on more than one blocklist that multi.surbl.org refers to, then the score is incremented by the score increment value for each list it appears on.

This plug-in will work "out of the box" with the default settings.

More than one instance of SURBL can be used should you wish to check more than one blocklist, by creating a copy of the plugin with a different name, e.g. "uribl". In this instance, it will refer to the config file at /etc/mailstripper/uribl.config.

Local Blocklist Plug-In

The Local Blocklist plug-in implements a locally-managed blocklist that is run in addition to the main downloaded blocklist file. Although the main blocklist file is editable, any updates will cause local modifications to be lost. The Local Blocklist plug-in avoids that by maintaining a separate blocklist file, though stored in exactly the same format as the main blocklist as described on page 15.

For this plug-in to operate, it requires a blocklist to be stored at **`/etc/mailstripper/blocklist.local`** and without this it cannot (and will not) run.

It is important to note that the scores returned by the local blocklist plug-in do not override those from the main blocklist, instead they are handled in addition to them. Therefore, an entry in the main blocklist with, for example, a score of 2, would need a local blocklist entry of –2 to effectively cancel it out.

The Plug-In API

If you are not interested in creating your own plug-ins for MailStripper, feel free to skip this section.

1. MailStripper calls the plug-in to perform some scanning operation.

Data supplied to plug-in (as command line arguments):
* **`-scan`** (required)
* MailStripper version (*this is to allow some degree of future-proofing*)
* IP address of sending server
* Filename of temporary file holding unprocessed data
* Filename of temporary file holding ready-for-scanning data
* Filename of temporary file holding the SMTP envelope meta-data.

All plug-ins must accept these CLI arguments in this order. The plug-in is free to ignore the arguments which are irrelevant to it, but must be aware of their existence and position.

Return value to stdout:

A score modifier (-1999 <= value <= 1999), to be applied to the final score after all adjustments made (length, caps subject etc).

Alternatively, the returned score may be:
-2000: Treat message as ***whitelisted***.
2000: Treat message as ***blacklisted***.
3000: Treat message as ***blacklisted***, and if no plugin suggested whitelisting, add the sender IP to the IP blacklist.

Return values **MUST** be supplied as a string on the plug-in's standard output device (stdout).

A zero response value implies that it should have no effect on the score.

If the plug-in cannot find the files given (e.g. interface change in the future) or recognises that the version of MailStripper is too old to work with it, it should return 0, and it is recommended that it reports an error via the # mechanism (below) to say what's wrong.

In the event of a blacklisted response, an error to be sent to the sender MAY be included after the number, separated by a space character. If no message is supplied, a default error will be used.

When multiple plug-ins are in use, the results are added up. When one plug-in sends back a whitelist code, the whitelist overrides everything else (including a blacklist code).

IMPORTANT: Plug-ins *cannot* override whitelisting and blacklisting set by MailStripper's internal rules. Indeed, should a message be either white- or blacklisted by any internal rules it **MUST NOT** be assumed that any plug-in will be run.

Indeed, if more than one plug-in is in use, and an early one returns a whitelist code, then it **MUST NOT** be assumed that later plug-ins will be called. However, a blacklist response may be subsequently overridden by a whitelist response later, so a blacklist response **WILL NOT** prevent subsequent plug-ins from running.

The response is to be on one line **ONLY**.

However, as an aid for debugging and error reporting, lines starting with a `#` will be captured and placed in the MailStripper log file, and not count as the response line.

The plug-in **MUST** exit with a success code. Errors should be reported to the plug-in's own log file, or via the `#` mechanism.

2. MailStripper calls the plug-in to identify itself.

Data supplied to plug-in (as command line arguments):
* **`-id`** (required)
* MailStripper version (*this is to allow some degree of future-proofing*)

Values returned, each on a line on its own, in this order:
* Plug-in name
* Plug-in version
* Plug-in author/copyright details
* Yes/No – whether or not this plug-in is compatible with this version of MailStripper.

Examples:
Execute: `plug -id 1.4.0`

Return:

```
Example Plugin
1.0
(C)2005 Eridani Star System
Yes
```

Execute: `plug -scan 1.4.0 192.168.254.23 /tmp/in.12345 /tmp/pre.12345 /tmp/meta.12345`
(this is all executed on one line)

Return:

```
2000 Content violation
```

In this instance, the message would be blacklisted unless another plug-in requests the message to be whitelisted. The sending mail server will get the error line:

```
550 5.7.1 Rejected: Content violation
```

If return code 3000 is used instead, all the above takes place, and additionally the sender's IP address is added to the IP blacklist file, so future messages from that IP (unless the message is whitelisted by an internal rule) will be blocked before the message content is sent over the connection.

The greylist rule of the Honeypot file is observed, so known authorised relays won't get blacklisted and in such case the rejection behaviour is the same as a 2000 return code.

(Note that MailStripper doesn't actually use /tmp, this is purely for illustration without long pathnames.)

The Meta-data file consists of at least two lines:

Line 1:	Sender address according to the SMTP `MAIL FROM` command.
Subsequent lines:	Recipient addresses according to the SMTP `RCPT TO` commands.

This is to enable plugins to use the envelope data that is not recorded within the message body (e.g. bcc'd recipients).

APPENDIX A – FILE LOCATIONS

`/usr/sbin/mailstripper`	The daemon itself
`/usr/sbin/xmstripconf`	Configuration GUI
`/usr/sbin/mstripconf`	Text-only config tool
`/usr/bin/score`	Scoring analysis tool
`/usr/local/bin/avwrap`	Example *avwrap* script
`/etc/mailstripper/plugins/local_blocklist`	Local Blocklist plug-in
`/etc/mailstripper/plugins/surbl`	SURBL plug-in
`/etc/rc.d/init.d/mailstripper`	Boot-time start-up script
`/etc/logrotate.d/mailstripper`	Log rotate config file

`/etc/rc.d/init.d/mailstripper` may be installed as `/etc/init.d/mailstripper` or `/etc/rc.d/mailstripper` on some systems.

These must not be moved:

`/var/spool/mailstripper/`	Temporary file location
`/etc/mailstripper/badips`	IP blacklist file
`/etc/mailstripper/blocklist`	Blocked words file
`/etc/mailstripper/config`	MailStripper configuration file
`/etc/mailstripper/domains.allow`	Permitted Domains list
`/etc/mailstripper/honeypot`	Honeypot trap list
`/etc/mailstripper/recipients.allow`	Email recipients whitelist
`/etc/mailstripper/recipients.tag`	Email recipients taglist
`/etc/mailstripper/senders.allow`	Email address whitelist
`/etc/mailstripper/senders.deny`	Email address blacklist

APPENDIX B – END USER LICENCE AGREEMENT

1) By purchasing or evaluating MailStripper you agree to enter into the following agreement with Eridani Star System (hereafter: "Eridani") for the use of the MailStripper Pro (hereafter: "MailStripper") software package.

2) The installation of MailStripper in itself does not assume that you have accepted this licence and does not require you to. The purchase of a registration code for your installation or obtaining an evaluation licence, however, IS deemed as your acceptance of this licence. However, you download and test MailStripper at your own risk. Neither Eridani nor its affiliates will be held liable for any damages or incidents occurring from your trial of the product.

   In otherwords, you may freely download the software to determine whether the software is compatible with your system at your own risk. If the configuration tool runs, the MailStripper daemon will also run. Without the registration code, however, you will be unable to alter any of the settings from the defaults, and the daemon will not run without a valid configuration file being present.

3) If you wish to distribute the software package as part of any other software offering, free or commercial, please contact Eridani at `mailstripper@eridani.co.uk` - the software may not be redistributed without prior written agreement.

   The MD5 checksum of the current version of the package is available from the website and can be used to confirm whether your copy is genuine, especially if it came from a third party.

4) You accept that paragraph 3 does NOT grant you the right to modify the package and redistribute it. Such actions are considered to be a violation of this licence and as such will be acted upon.

5) You agree that your unique registration code MUST NOT be distributed. Distribution of this is deemed to be an act of piracy and will be acted upon. Registration codes found to be distributed will be deactivated.

6) Your copy of MailStripper may be licensed either for non-commercial personal use, or for commercial use.

   IF YOUR LICENCE IS FOR COMMERCIAL USE:
   You may only use MailStripper on one server. If you need it on more than one server, these need to be individually licensed.

   IF YOUR LICENCE IS FOR PERSONAL USE:
   You may install MailStripper on any number of machines that are your own hardware at your home.

   If you run a business at your home you need the commercial licence. Use of email to conduct online shopping is considered personal use unless it is in conjunction with running a business.

7) You agree that no spam scanner can be 100% accurate, and that some legitimate mail might get accidentally caught as spam. It is your responsibility to ensure that the nominated spam file is periodically checked for false positives. You also accept that on occasion some spam might not get detected. For this reason the configuration tools allow you to tailor what is seen as spam. Although the software will permit setting the spam file to `/dev/null` Eridani does NOT recommend this, as any false positives would then be lost without trace. Eridani hereby disclaims any responsibility for such misconfigurations.

8) If you use an anti-virus scanner in conjunction with this product, you agree to abide by the licensing terms of that anti-virus scanner, including payment of any appropriate licence fees to the vendor. You understand and accept that Eridani has no relationship with any anti-virus supplier, and MailStripper only makes use of an anti-virus scanner if you choose to install and configure it. You also agree and understand that any email found to have a virus infection will be quarantined as a whole and, depending on the configuration of MailStripper, the message may be deleted in its entirety. You accept that Eridani hereby disclaims any liability for data lost. Eridani bears no responsibility or liability for computer viruses or the performance or lack thereof of any third party products.

9) You accept that Eridani hereby disclaims any liability for MailStripper's use as an open mail relay. When properly configured this will not be an issue - for the reason that, should your MTA be run on the same server as MailStripper, the use of the localhost interface (127.0.0.1) for communication is strongly discouraged and the Permitted Domains file should be properly configured.

10) You agree that Eridani cannot and will not be held liable for any loss or damage caused by downloading and deploying MailStripper however that data loss might arise, including but not limited to malicious actions of other MailStripper users.

11) You agree that, irrespective of your location, the laws of England govern any dispute you may have with Eridani over MailStripper Pro, and both you and Eridani submit to the exclusive jurisdiction of any competent Court in England.

12) You acknowledge that Eridani recommend that you keep your registration code safe, as you will need it again should you need to reinstall your system. Also, as your internal registration key is connected to your hardware, you may need to re-enter it should you upgrade or alter your hardware configuration. Until the key is re-entered, MailStripper will continue to accept incoming SMTP connections but will not filter for spam or viruses, and you will not be able to alter any of the settings. Eridani is not liable for any registration codes that have been lost, misplaced or misused.

13) You acknowledge that, when an evaluation licence expires, MailStripper will continue to accept incoming SMTP connections but will not filter for spam or viruses, and you will not be able to alter any of the settings.

14) Your licence entitles you to use of the software, in that it does not govern the use of any ancillary services or plugins other than those plugins supplied as part of the MailStripper package. In particular, it does not entitle you to the subscription blocklist service without a separate subscription.

15) You remain bound to this licence until terminated. We may terminate the licence given reasonable grounds to do so including, but not limited to, payment charge-backs. This licence is also terminated by the expiry of any evaluation licence.

16) MailStripper must be permitted to contact Eridani’s web server at port 80. Failure to do so may result in interruptions in service.

17) Eridani reserves the right to implement any automatic licence verification mechanism. Such mechanism will involve a challenge/response mechanism, and no information besides your licence code and random data for providing secure checksums will be sent to Eridani. The software will only deactivate itself should it be directly instructed from this that the licence has been terminated.

18) Any provision of any contract or licence agreement with Eridani which is or may be void or unenforceable shall to the extent of such invalidity or unenforceability to be deemed severable and shall not affect any other provision of any contract or licence agreement.

19) No waiver or forbearance by Eridani (whether expressed or implied) in enforcing any of its rights under any licence or contract shall prejudice its right to do so in the future.

20) If you do not agree to these terms and conditions, do not use, store, or download this or any related software.

In respect to clauses 16 and 17, MailStripper will only send the following information to us: Your licence code (for verification and entitlement purposes), and the process ID making the connection for making a more robust checksum for the challenge/response communication. Eridani will never be sent any details of your system, and to reinforce this the MailStripper licence does not forbid you from running a network packet sniffer on any communication by MailStripper, internally or externally. In the case of remote-deactivation, we have only ever had to do this once, when a stolen credit card was used to purchase the licence.

MailStripper is statically linked to and therefore includes the PCRE library:

Please note that this licence specifically only applies to the statically-linked PCRE component, and not MailStripper as a whole.