Spam Lists To Be Spam = 1

In order to be treated as spam, the sender of the message must appear in at least this number of "Spam Lists" to be treated as spam. Previously this setting was fixed at 1, so that a message appearing in a single "Spam List" would be treated as spam.

Every word in every processed message is added to the Bayes database, so it can grow

`Wait During Bayes Rebuild = no`

The Bayesian database rebuild and expiry may take a few minutes to complete. During this time you can either wait, or simply disable SpamAssassin checks until it has completed.

`Incoming Work Dir = /var/spool/MailScannel/incoming`

Set where to unpack incoming messages before scanning them This can completely safely use tmpfs or a ramdisk, which will give you a significant performance improvement.

NOTE:

The directory (or a link to it) containing all the Sophos *.ide files. This is used by the "sophossavi" and "sophos" virus scanners, and is irrelevant for all other scanners.

`ClamAVModule Maximum Recursion Level = 5`

ClamAVModule only: the maximum depth to which archives will be unpacked. If you set this too high you are inviting denial-of-service attacks, but you should set it deep enough that normal users will not pack archives deeper than this setting.

Allow IFrame Tags = disarm

Do you want to allow <IFrame> tags in email messages? This is not a good idea as it allows various Microsoft Outlook security vulnerabilities to remain unprotected, but

Normally, shell environment variable names are all upper-case. They can be of the form $HOSTNAME or ${HOSTNAME}. The latter format is used when it is not obvious how to separate the name of the environment variable from any text following it. It is usually safer to use the ${} form, and cannot do any harm. The string "\n" which will be replaced by a newline

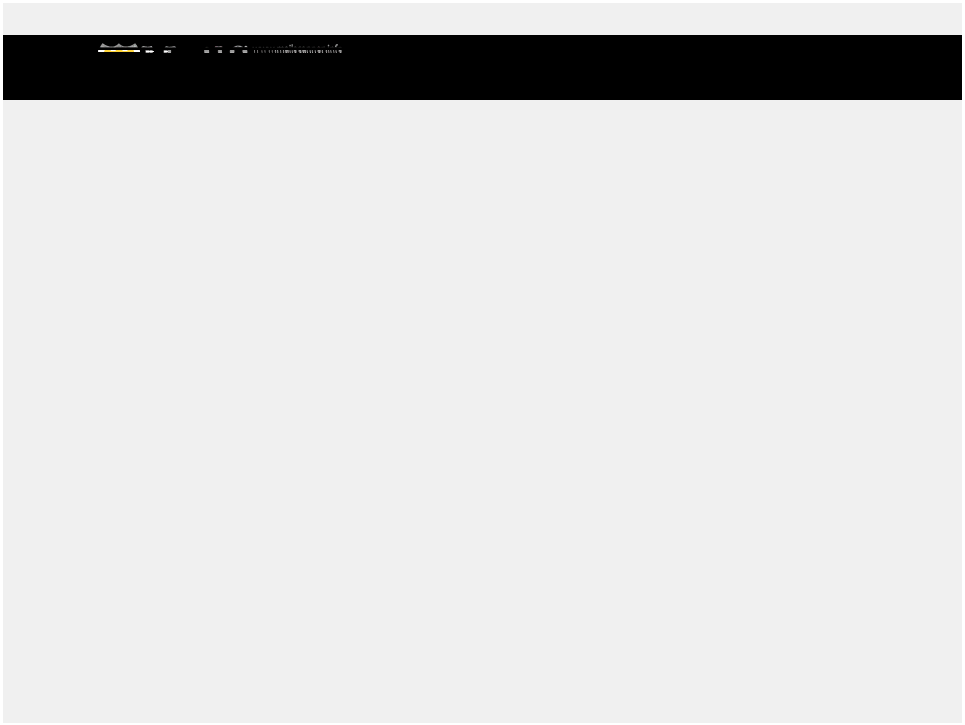
Hostname = The %org-long-name% (${HOSTNAME}) MailScanner

This nicely demonstrates the use of both %variables% and shell environment ${variables} in order to pr

using a ruleset for the “SpamAssassin Required Score” option, then the number “8” would be suitable for the right-hand side of the rule. If the ruleset was for the “Spam Checks” option, then the only valid right-hand sides would be “yes” or “no”.

The contents of the right-hand side is simply the setting that you want to apply to the configuration option, when the condition in the rule matches the message.

Rulesets cannot be nested. The only nesting allowed is when



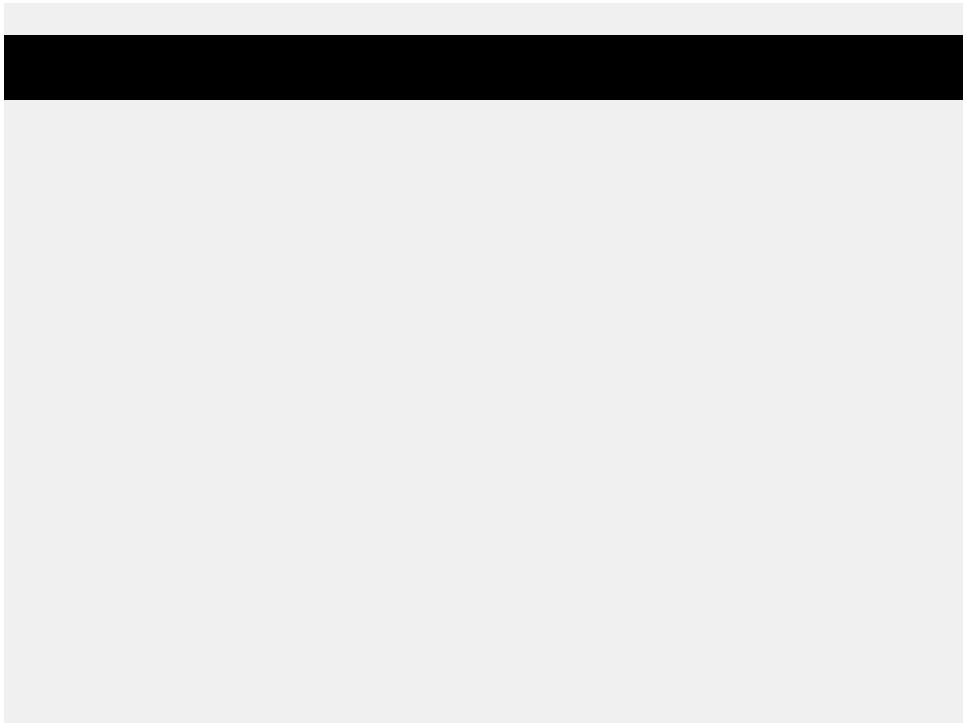
This example is rather more complex, but is one of the most common requests from the MailScanner user community.

Most of your customers are happy with the filename checking provided by the default allow/deny tests supplied with MailScanner.

However, 2 of your customers have slightly different requirements.

- One of them wants to ban its staff from sending out Zip archives ending in ".zip", but allow all Microsoft Word ".doc" documents, regardless of the rest of the filename. Their domain name is domain1.com.

- Another customer wants to allow its support staff to send out ".ini" files, which are used by the software they sell and they need to be able to send these files to their own clients to assist them. Their domain name is domain2.com.



There are situations in which you will want to user a content checker you have written yourself, or you want to use some other content checker that is not directly supported by MailScanner.

The “Generic Virus Scanner” provides a means whereby you can do your own content checks. There is a “generic-wrapper” script provided for you to start and run your content checker. You will have to write your own parser, which should go into MailScanner/SweepViruses.pm, to process all the output from your checker.

If your checker needs any form of regular updates, such as new virus signatures, then you should customise the “generic-autoupdate” script as needed.

The “Generic Spam Scanner” gives you the ability to easily implement support for unsupported spam scanners such as CRM114 and dspa

Input:

1. SMTP connection IP address
2. Envelope sender
3. List of envelope recipients, one per line
4. 1 blank line
5. The whole message, including all the headers and the body in rfc822 format

Output:

1. Floating point or integer giving the spam score of the message
2. 1-line report giving a brief summary of the results of the program

Use Custom Spam Scanner = no

If this is set to yes, then the custom (or “generic”) spam scanner will be applied to the message. Base your code on the GenericSpamScanner.pm file in the