
AntiSpam Guide

8.3.8



Contents

Introduction	1
<hr/>	
How It Works	3
<hr/>	
MIAS Scoring System	6
Actions	6
Bypassing MIAS	7
Spam Folders	7
Processing Flow Chart.....	8
<hr/>	
Administration	12
<hr/>	
General configuration.....	12
MIAS action configuration	14
Other Configuration	15
Statistics	17
<hr/>	
Configuration Files	18
<hr/>	
Bypass Files	19
MIAS Properties in Detail	19
<hr/>	
Spam Assassin	21
<hr/>	
How It Works	22
SA Scoring System	22
Reporting Function.....	23
Additional Filters	24
Administration.....	25
Configuration Files	27
MIAS Properties in Detail	28
<hr/>	
Sender Policy Framework and Sender Rewriting Scheme	29
<hr/>	
Administration	30
<hr/>	
DomainKeys	31
<hr/>	
Administration	32

Greylisting **35**

How It Works	36
Administration.....	38
Configuration Files	40

Challenge Response **41**

How It Works	42
Administration.....	46
Configuration File.....	51

Bayesian Filters **52**

How It Works	53
Administration.....	56
Configuration Files	58

Body & HTML filters **59**

How It Works	60
Body & HTML filters.....	60
Charset filters.....	60
Administration.....	62
Configuration Files	66

Content Filters **67**

How It Works	68
Administration.....	70
Content Filters Export / Import.....	72
Configuration Files	73

Black and White Lists **73**

How It Works	74
Administration.....	76
Configuration Files	78

Other **78**

Tarpitting.....	79
How it works	79
Administration	79
Configuration files	81
DNSBL	82
How it works	82
Administration	82
Configuration files	83
Miscellaneous	85
HELO/EHLO.....	85

Delayed SMTP session processing.....86

Troubleshooting 87

Engine Logging88
 Actions88
 Reasons90
 SMTP Test Tool92
 FAQs.....94
 Why Is Spam Message in My Inbox Folder?95
 Why Is Legitimate Message in Spam Folder?.....96
 How to check the Automated General Spam Reference Base Update.....96
 How to Enable SPF97
 How to customize Challenge Response messages?98
 How to Stop Spammers Using ESMTP and Demo/Known Accounts.....99
 How to Use Merak as your AntiSpam and AntiVirus Gateway for MS Exchange Server?..... 101
 How to enable spam folders for selected users 105
 How to bypass all local messages from Spam Scanning ?..... 106
 How to Create a Spam Trap..... 109

Feedback 111

Index 112

CHAPTER 1

Introduction

Spam by e-mail, simply said, is unsolicited sending of identical (or nearly identical) messages to thousands (or millions) of recipients without the permission of the particular recipients. Since receiving spam brings unnecessary load on the mail server, and ultimately annoys the end customer – the mailbox owner, there have been initiatives to introduce a variety of complex procedures into Merak Mail to help filter these unsolicited messages out.

These procedures are divided into two groups. One would contain methods based on the message content analyzation, the second group is based on sender verification. Whereas sender verification is bound to the credibility of some servers and to unified sender signatures which are mostly simple methods (based on innovative ideas), message content analyzing methods are rather complex and mostly based on statistics.

It's an apposite comment that content analyzation never leads to an exact answer - the result is always just a probability. For example, we can get a result "93% percent probability that the message is spam", and it's up to us to determine what we will do with such a result. Actually even the number is just a relative value and we have to configure the antispam engine so it has as few false positives (a legitimate message that was mistakenly marked as spam) as possible.

Successful configuration of an antispam engine will decrease the number of false positives but will introduce other problems. If the engine doesn't miss any legitimate messages, it could be too loose so it lets too many spam messages reach your users' mailboxes because they were not recognized (so called false negatives). In this case, we don't lose an important messages, but the load caused by spam filtering was absolutely wasted.

Moreover, this fight seems to have no end because every single antispam method that has been announced in the past has brought a reaction from organizations that found a business in sending spam (spammers). Therefore the efficiency of each antispam technique slowly decreases through its use.. The ratio of the count of spam messages to the total number of messages received continues to grow each year.

Merak Instant Anti-Spam is a server based solution that was designed to protect all server users from unwanted Spam e-mail.

To protect users from Spam, the Merak Instant Anti-Spam Engine combines several techniques:

- § SpamAssassin and compatible content filtering
- § Greylisting
- § Sender Policy Framework and Sender Rewriting Scheme
- § DomainKeys
- § Challenge/Response System (confirmed automated senders white listing)
- § Bayesian Filters with automated Reference Base update and "self learning" features
- § White & Black Listing
- § Sophisticated programable Content Filters

§ and many more antispam features

All components intelligently work together to deliver genuine e-mail to the recipients Inbox and received Spam e-mail to the user's Spam folder. Remember that every technique is designed to recognize/catch different kinds of spam messages. Therefore every single method has its pros and cons, none of these methods are all-powerful and the best way is to use all of them at once to reach the most accurate result. Administration of all components can be performed at one place and even though it offers high degree of flexibility for advanced users, it can bring pretty good results when it's simply turned on.

The Inbox and Spam folder can be read by:

§ Any mail client with IMAP access (for example Microsoft Outlook or Outlook Express)

§ Integrated Merak WebMail access

§ Instant Messenger Client - Just Another Jabber Client (JAJC)

The Spam folder obviously contains Spam e-mail and it is usually not necessary to watch that queue, however, if a genuine e-mail is placed in the Spam folder (a fact usually called a false positive), it can be easily moved to the Inbox and vice versa.

The content of a user's Spam folder is auto-deleted after a specified number of days.

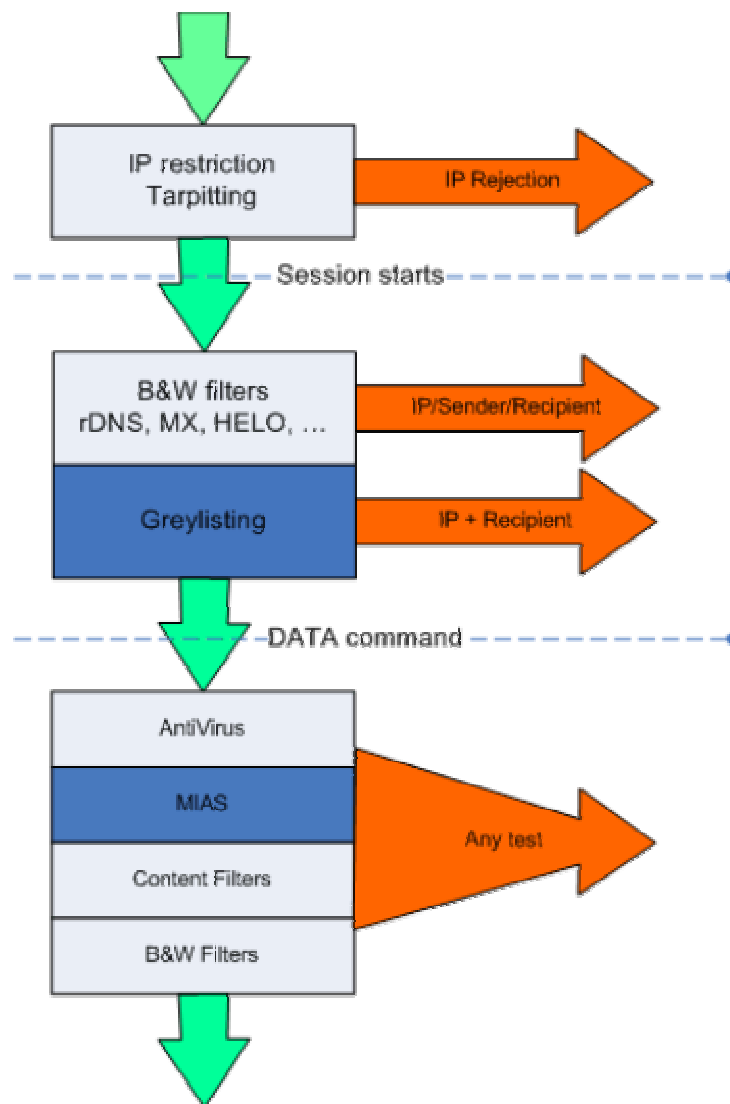
Using all of the above filtering methods, Merak Instant Antispam can achieve near 100% accurate spam identification. It does not require any special software on your users workstations and the entire system can be used without any user maintenance. For tracking purposes, detailed antispam logging is also available.

CHAPTER 2

How It Works

In this chapter we will try to describe the entire Merak Instant Anti-Spam (MIAS) engine functionality. We will show, where it is placed in the Merak message processing pipeline, what are the most important settings and we will also point to all related topics that are not internal parts of MIAS, but that should be mentioned when we are talking about fighting with spam.

First, take a look at the following diagram. It schematically shows the message processing pipeline.



As you can see, there is quite a variety of tests, but divided thematically into three groups. The first group contains tests based on a client's IP and is applied before any SMTP command can occur, thus increasing server performance when properly applied. Basically it covers tarpitting and IP restrictions defined for SMTP service.

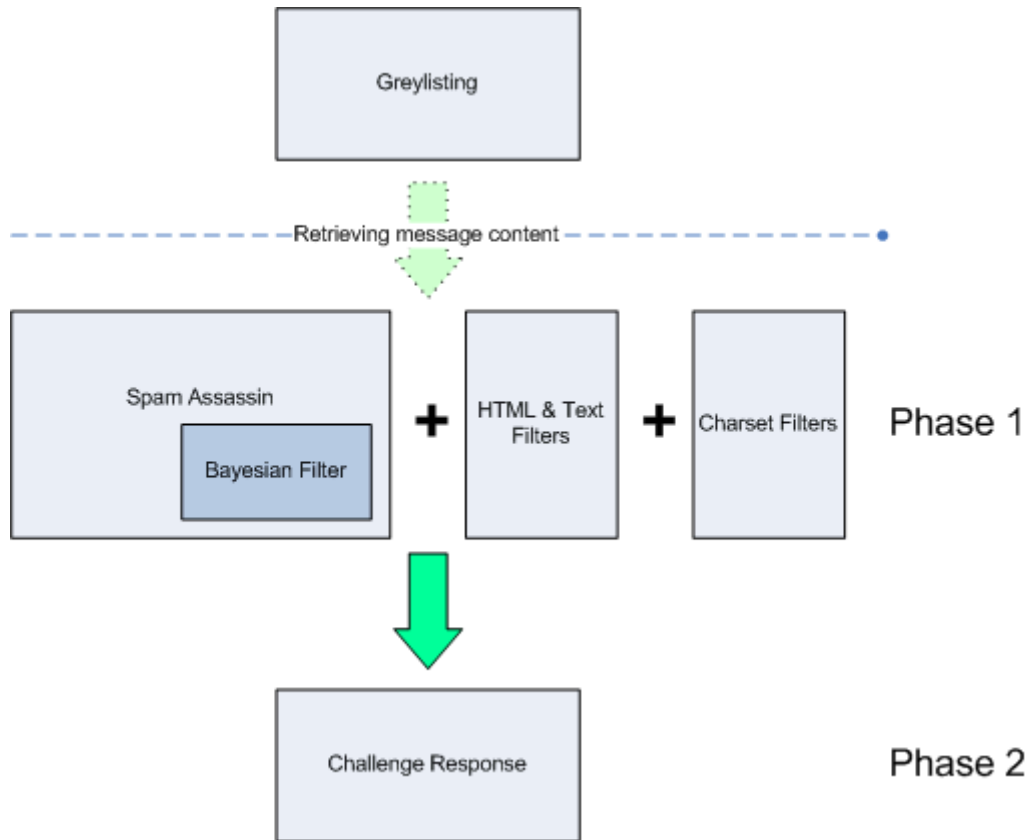
The second group of filters contains several filters based on the client's IP, and the sender's or recipient's address. The most important are B&W filters (or the appropriate portion of them that is not based on message content), but there are also rDNS and MX checks etc. B&W filters will have a special section here in this guide because they are one of the most important filtering practices. All the tests in this group are applied before the DATA command is used in the SMTP session. This fact makes it the right place for any possible rejection that can be done without knowledge of message content because it saves server bandwidth and even CPU load caused by unnecessary filter processing. Notice that even the greylisting takes a place here (that's because greylisting is not dependent on message content).

Filters from the last group are applied once all the message data is obtained. By default, Anti-Virus is applied as the first one, followed by MIAS. All Anti-Spam components but greylisting are run from this point. The remaining B&W filters are also performed when all data is received. They can be used for rejecting messages just as like marking it as spam or genuine. The last filter group is content filtering that can (as like as B&W filtering) reject a message or mark it as spam or genuine (cancel previous rejection/spam flag). Please, notice that neither of these B&W filtering rules nor content filters can prevent a message from being processed by MIAS, but they can remove a previously obtained spam flag or even more they can perform some action on the basis that the message is spam or not.

The MIAS engine itself is a complex object. It contains several components and each component has its purpose:

- § Spam Assassin
- § Challenge Response
- § Bayesian filtering
- § HTML filters
- § Charset filters

In fact, Bayesian filtering is an integrated part of Spam Assassin, even though it has its own configuration. The following diagram displays relations between all mentioned components including Greylisting.



As you can see, we have divided the Anti-Spam engine processing into two phases. In the first phase, the Spam Assassin with Bayesian filtering, HTML and charset filters are run, whereas the second phase contains just the Challenge Response mechanism.

MIAS Scoring System

The first phase is using the scoring system. That means - a message is checked by several tests and each test adds a point score to the message depending on the test result. All these points are summed up and in the end, if it reaches some limit, the message is considered spam. There are several components that contribute to a message's number. The resultant number is made up as a sum of the *Spam Assassin* result, the *HTML filters* (see "Other" on page 78) result and the *Charset filters* (see "Other" on page 78) result. We can use following equation:

$$\text{TotalScore} = \text{SAScore} + \text{HTMLScore} + \text{CharsetScore}$$

When the total score is computed, it's compared with defined limit(s) and the MIAS engine decides what will be done with the message:

- § When the computed score is lower than the specified limit, the message is passed to the Challenge Response mechanism and the rest of the pipeline.
- § When the computed score is equal or greater than the specified limit, it is considered spam and some defined action is performed. The message could still be passed to the Challenge Response, depending on the action taken. (see below - Actions)

If the message is passed to the rest of the pipeline, the second MIAS phase takes place. It contains the *Challenge Response* (on page 41) mechanism and works much differently than the first phase. It doesn't use the scoring system, but uses results of the first phase as input. Strictly speaking it considers spam suspicion.

Actions

As it was already mentioned, MIAS can perform several actions depending on the message's score. These actions are:

- § Mark message as spam
- § Quarantine message
- § Delete message

The Anti-Spam engine can even do nothing - that's the default action, but we surely want to use the spam marking action at least. Each of these actions can have an associated score with it. When this score is reached the action is performed. The mark message as spam action does nothing more than setting a spam flag for that message. This is commonly set to some lower value (like 3.0) and by setting this you enable your users to manage such (considered-to-be-) spam messages on their own. When quarantine message action is used, the message is resent to some address that you can specify (some spam account you use for statistics) and is deleted from the user's mailbox. Using this method, the user is unable to manage the message, as he won't even know he received it. Some higher score (6.0 or more) is recommended for the delete message action. Here the name is self-explanatory - the message is simply deleted.

The highest score a message can reach is 10.0, which is a fine score for immediate message deletion if you don't receive nasty spam. But don't forget, all of these tests are based on statistics, so even a highly scored message could be a legitimate e-mail (so called false-positive) and a zero scored message could be just unrecognized spam (false-negative). There's no limit for negative values, but common legitimate messages generally score between -2.0 and 1.0. (Scores for advertising messages will be surely higher.)

Bypassing MIAS

It's always useful to prevent messages from being processed by Anti-Spam when you don't need them to be checked. There are several ways to accomplish this:

- § use *bypass file* (see "Configuration Files" on page 18)
- § enable trusted IPs bypassing
- § create a *B&W rule* (see "Bayesian Filters" on page 52)
- § create a *Content Filter* (see "Content Filters" on page 67)

The most common way is using a *bypass file*. It can contain an email address or domain name of both sender or recipient of a message that should be bypassed. This is used when you have some friendly domains. Another similar way is using the "Bypass trusted IPs and authorized session" feature. It's even more simple than the *bypass file* because you just have to turn this on and it will work without any further management. The philosophy of this is very simple - once a sender is considered trusted, it's unnecessary to check his messages for spam. We highly recommend that you use this option, especially if you are about to use *greylisting*.

Both B&W rules and Content Filters are more complex – as they can perform a special test (considering message headers and content) and mark the message as genuine. Whereas a B&W rule is checked before Anti-Spam processing and thus can prevent MIAS tests from being applied, Content Filtering is done after Anti-Spam and so it can only reset the message flag, but all MIAS tests have already been made.

Spam Folders

Spam messages can be automatically moved to a special folder in each user's mailbox so that the user can prevent downloading of such messages from the server to his email client or simply to have all messages sorted. This special folder is called the Spam folder and the sorting is performed implicitly on the server when the message is received. This folder is created automatically for each user - during the first spam message storage.

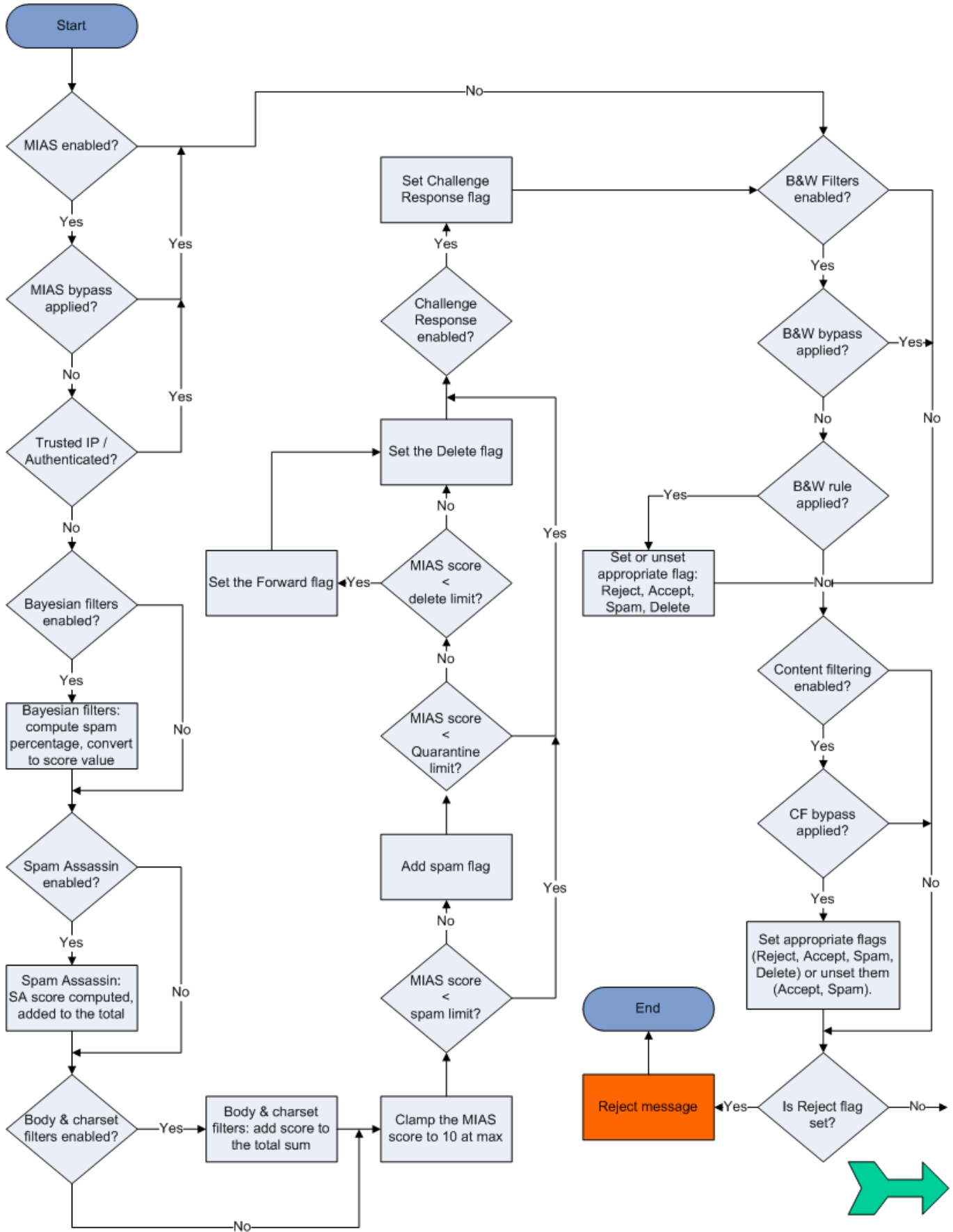
Moreover, Merak has a feature "Delete Spam Messages From Spam Folder When Older Than (Days)" which will allow you to specify the number of days that messages will stay in the Spam folder. When the specified count of days passes, the spam message is automatically deleted. If this feature is disabled, spam e-mails will remain in the Spam folder permanently and you along with your users will have to delete them manually.

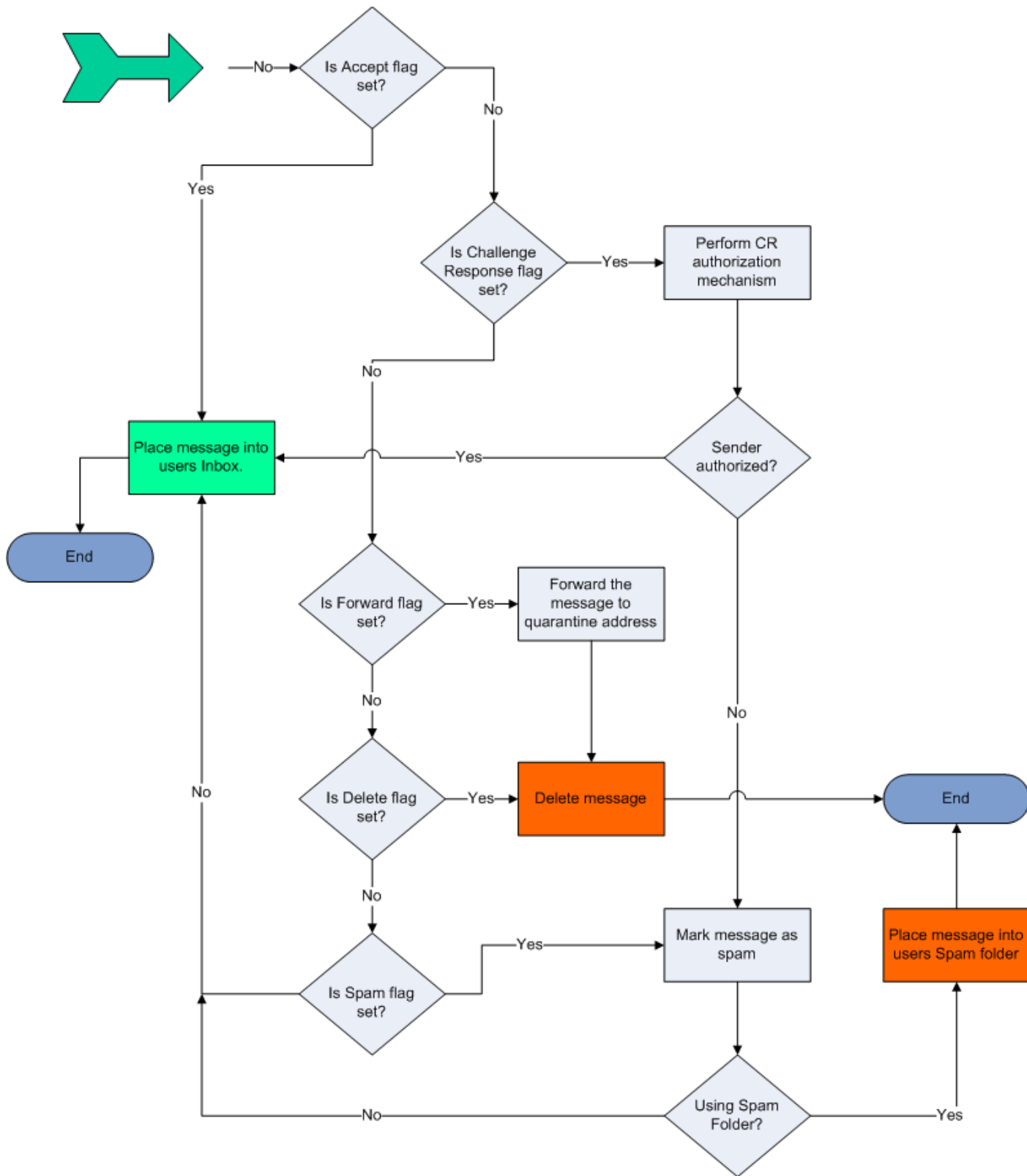
Notice, that the Spam folder is actually just a common folder in a user's mailbox and thus it can't be checked from your mail client using the POP3 protocol which does not support folders. So if you want to use the spam folder, you have to either use the IMAP protocol (and IMAP mailbox type) or access your mailbox through WebMail from time to time.

For activation of spam folders on your server, use the Windows administration console (or similar web based interface), where you can find the setting under the Anti-Spam section in Action tab.

Processing Flow Chart

Now it is good time to show the whole antispam processing mechanism in a flow chart. It shows all the relevant processes and decisions Merak performs when it is processing the message content (greylisting is missing here because it is applied before message data is received).





CHAPTER 3

Administration

Merak Instant Anti Spam engine may be configured in several ways. The most easy to use way is with the Merak Administration Console, but it's a solution available only to Windows users. Alternatively, a similar interface is usable through your internet browser. The following screen captures have been taken from the Windows Console, but all the options described here are accessible from the web based interface as well. If you prefer to configure Merak by editing text files, please, read the *Configuration Files* (on page 18) section.

General configuration

General | Action | SpamAssassin | Greylisting | Challenge Response | Bayesian Filters | Body | Other

General

Active Processing Mode...

Field	Description
Active	Activates the complete Merak Instant Anti Spam engine.
Processing Mode	Here you can specify for which domains (or accounts) Merak Instant AntiSpam will monitor.

Updates Schedule

Enable At:

Su Mo Tu We Th Fr Sa Update Now

Options in the "Updates Schedule" section allow for "hands - free" maintenance of the "Reference Base". The "Reference Base" is used by the Bayesian filters for accurate Spam recognition.

The main Reference Base is maintained by our staff and ensures the optimum Bayesian Filter performance for most users. It is downloaded from the location, which is given by the :

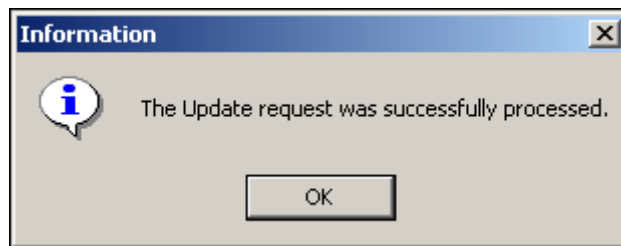
§ SpamUpdateURL=<http://www.icewarp.com/update/spam.xml>

§ parameter in the /Merak/Spam/Spam.dat file

It is currently located at: <http://www.icewarp.com/update/spam.dat>, but it may differ in future. Always check the content of the file: spam.xml

In the fully licensed MIAS, you can set the time and date that your server will check for the latest updates. New updates are automatically downloaded and installed transparently. You can also use the Update Now button.

When you press the "Update Now" button, Merak will send an update request and you should receive the following response:



The amount of time that it actually takes to update depends on the speed of your connection.

The average file transfer is roughly 800kB.

The update will proceed only if the date and version of your current Spam Reference Base is earlier than the current downloadable version.

The Update feature only replaces the "General Spam Reference Base", maintained by IceWarp. Your own indexing base maintained by you will remain unchanged.

Important note: The current Instant Anti-Spam solution combines many different techniques. For the user, indexing is not so important to achieve proper Spam recognition. It is normally not necessary for you to "teach" your system through indexing. You should simply set the system to automatically download the "Reference Base".

Information	
Last update date:	2004/12/29
Last update version:	7.7.5
SpamAssassin engine version:	3.0 (1.3)

The last section under the General tab shows the information about the current General Spam Reference Base you have downloaded on your server.

MIAS action configuration

General	Action	SpamAssassin	Greylisting	Challenge Response	Bayesian Filters	Body	Other	
Action								
<input checked="" type="checkbox"/>	Score required to classify message as spam:							3.00
<input checked="" type="checkbox"/>	Score required to quarantine message:							7.00
<input type="checkbox"/>	Score required to delete message:							10.00
<input checked="" type="checkbox"/>	Add text to Subject of spam message:	[Spam %%SpamReason%% %%SpamHits%%]						
	Quarantine email address:	spam@doc.icewarp.com						

Field	Description
Score required to classify as spam	The default score used to classify a message as Spam is 3. This value has been calculated by our own statistics. You can change it to some other value that might better fit your needs. A higher value will cause the Spam filter to be less able to recognize Spam messages, but there will be fewer false positives and vice versa.
Score required to quarantine message	When this score is reached the message is quarantined. I will be sent to the e-mail address specified in Quarantine e-mail address . It is very similar to the delete action.
Score required to delete message	When a message reaches this score, it is automatically deleted. Be very careful with this option as it will delete regular email when the level is set too low.
Add text to Subject of spam message	<p>Activates Spam Subject Marking Mode.</p> <p>For each message recognized as Spam (or marked as Spam from another filter, e.g. Content Filter), this filter will add the text [Spam] at the beginning of the original subject.</p> <p>The text "[Spam]" can be replaced with any other string you desire.</p> <p>If you want to see which filters resulted in the message being recognized as Spam, simply use the "%%SpamReason%% " system variable. (<i>See /merak/examples/variables.dat for the list of the all system variables</i>)</p> <p>If you need to place the word text "[Spam]" or any other word at the end of the subject line instead of the beginning, simply use the "Content Filter: to extend the X-headers for reporting purposes.</p> <p>Use this option if you have users that would like to use their regular Email Client such as Microsoft Outlook, Outlook Express or Eudora. Simply marking Spam email with the word "[Spam]" in the subject line significantly increases user productivity by allowing them to identify Spam e-mail and review/delete them..</p>
Quarantine email address	Messages that reach a score for quarantine are sent to this address. This field supports multiple addresses separated by semicolons.

Spam Folder

Place spam messages under spam folders

Delete spam messages from spam folders when older than (Days):

Integrate spam folders with IMAP accounts (Folder name):

Field	Description
Place spam messages under spam folders	<p>Activates the Spam Folder Mode.</p> <p>Enable this option if you would like to create a special "Spam Folder" for the user that allows the Instant Anti Spam Engine to place all Spam e-mail automatically into this special folder.</p> <p>With this feature is enabled, Merak will place only genuine e-mails in the user's Inbox while separating e-mail identified as Spam into their Spam Folder.</p> <p>You can also define a "Spam Administrator" that can maintain the "Spam Folders" for one or more users. Using these automatic and manual techniques, you can truly achieve near 0 false positives. This is a significant time-saving feature for VIP users. For example, an office assistant can be the "Spam Administrator" for their boss.</p> <p>The "Spam Folder" can be viewed and/or manipulated via Web Mail or an Instant Messenger Client or by any regular client if it is set as an IMAP or POP/IMAP account.</p>
Delete spam messages from spam folders when older than (Days)	<p>After the number of days specified, the messages in the Spam folder will be automatically deleted.</p> <p>A user can move messages at any time from the Spam folder to their Inbox or vice versa.</p>
Integrate spam folders with IMAP accounts (Folder name)	Enter the name of the IMAP folder that will be used for Spam e-mail collection.

Other Configuration

General | Action | SpamAssassin | Greylisting | Challenge Response | Bayesian Filters | Body | Other

Logging

Debug

Summary

Here you can set the type of logging you would like to use with your Integrated Anti-Spam.

No Logging

If both checkboxes are off, the logging is switched off.

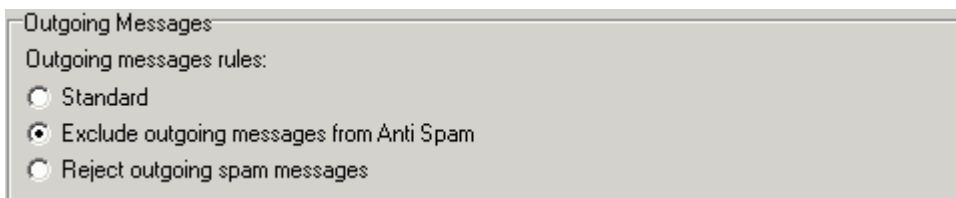
Debug Logging

The most detailed logging will be used showing all AntiSpam proceedings

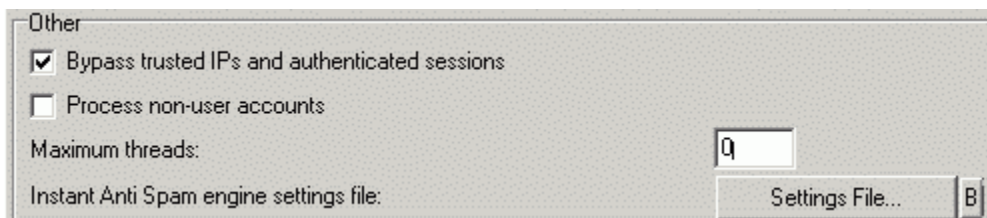
Summary Logging

Summary type logs only the general summary information and status of a Merak Integrated Anti-Spam.

The log files are located in the default location 'Merak\log\antisпам\'



Field	Description
Standard	In a Standard mode, all outgoing messages are checked for being Spam by the Instant Anti-Spam engines.
Exclude outgoing messages from Anti Spam	If you are certain that your users on your Merak Mail Server are not sending Spam, you can select this option. Messages sent out from the server will be never be marked as Spam in this mode.
Reject outgoing spam messages	This option ensures that outgoing messages that are identified by Instant Anti-Spam as Spam are rejected and not sent. Select this mode if you want to prevent Spam messages from being sent out from your server.



Field	Description
Bypass all trusted IP addresses	All IP addresses located in the Relaying from field or relay.dat will be bypassed for the whole anti spam engine. This is a recommended setting.
Process non-user accounts	Enable AntiSpam and check for all entities created within mail server like mailing lists, static routes, notifications, catalogues,... The MIAS engine is executed and it is up to mail server or spam administrator to use Content Filters or Black and White Lists to ensure the desired behavior.
Maximum threads:	Specify and limit the number of threads used by the whole MIAS. (For example, you do not wish your MIAS to take 100% of your CPU utilization.)
Settings File	This file contains the values of the options related to the Antispam engine. It is only intended for developers.

Statistics

Once you have the MIAS engine configured and working, you will surely welcome some overall information about all MIAS components or other filtering methods. You can check it out in the Statistics panel under the Status section.

☐ Messages

Received: 554
Sent: 7049
Failed Delivery: 640

Spam: 20 (3.61%)
Virus: 0 (0.00%)
Content Filter: 0 (0.00%)
B&W Filter: 1 (0.18%)
External Filter: 0 (0.00%)
Tarpit: 525 (94.77%)
DNSBL: 38 (6.86%)
GreyListing: 639 (115.34%)
Total Filters: 1223 (220.76%)

Configuration Files

Anti-Spam administration using configuration files is a bit harder because you need to have more knowledge about the purpose of many properties and also the file syntax. Without this knowledge it's rather dangerous to do the administration this way.

There are several files used. Every anti spam technique mentioned in this guide can be configured by editing some file, you can find precise descriptions in the appropriate sections of this guide. Here we will discuss the main MIAS configuration file - its syntax and every property used globally. Then you can learn the syntax of the bypass files used by Merak. Even if there are a number of bypass files, all of them use the same syntax. Here we will describe the syntax and in the appropriate chapter, you can find which file to edit for a particular feature or MIAS component.

MIAS Config

The main configuration file for MIAS is spam.dat located in the <MerakInstallPath>\spam\ directory. It contains the configuration for all MIAS components except the Spam Assassin which uses its own files with the same common file format. The structure of spam.dat file is very simple: Each property is written on a separate line, which consists of the property name and associated value (both in plain text) separated by an equivalence character. For example, setting text that is being added to the subject of each spam message follows:

```
AddSubjectString=[Spam]
```

We should also discuss the syntax features in detail, so here are some facts you can find valuable:

- § The order in which all properties appear in the file does not matter.
- § When some property is missing, a default value will be used.
- § If the property appears more than once in the file, the last occurrence is used.
- § When a property is of a Boolean type (true/false), 1 signifies true and any other value is translated as false.
- § Whitespaces (spaces and tabs) are ignored. Thus you can't create a string variable starting with a whitespace.
- § Any line that does not start with a known variable name is ignored.
- § There are no native comments; however you can simply create a comment line that does not start with a known variable name as noted above..

There are no real restrictions and the file syntax (that is used for most configuration files in Merak) is very forgiving. For a complete list of supported variables please consult the end of this chapter.

Bypass Files

Bypass files in global are used to define which messages should be prevented from scanning/processing by any particular filter. It's based on sender/recipient matching with any entry in the bypass file. The syntax is the same for all such files:

- § One entry is an account (john@domain.com), address (mail.domain.com) or IP (127.0.0.1).
- § Every entry (account/address/IP) is placed on a separate line.
- § Asterisk character (*) can be used; it represents any count of any characters, thus could be used for specifying IP mask (127.0.0.*) or domain bin (*@domain.com) etc.
- § When specifying an IP address, CIDR notation can be used (127.0.0.0/24).
- § Line comments are allowed. Sequence of characters “// ” (slash, slash, space) starts a comment that ends with the end of line.

The sender, recipient and sending IP are now compared with any single entry in an appropriate bypass file. If any of the mentioned items are contained in any file entry, filter processing is bypassed for that message.

MIAS Properties in Detail

Here we will list and briefly describe some properties that can be used in the spam.dat file. Further properties can be found in the Configuration Files section of any particular method description.

Variable	Type	Description
SAScore	Boolean	Set this to 1 if you want to enable the spam flagging function for messages that reach some specified (see below) score.

SAScoreValue	Float	This variable is considered obsolete. Don't use it. Use required_hits property in local.cf file instead.
SAQuarantine	Boolean	Set this to 1 if you want to enable the quarantine function for messages that reach some specified (see below) score.
SAQuarantineScore	Float	Minimal score required for quarantining the message.
QuarantineAddress	String	mail address to which quarantined messages are being sent.
SADelete	Boolean	Set this to 1 if you want to enable spam deletion for messages that reach some specified (see below) score.
SADeleteScore	Float	Minimal score required for spam message deletion.
SpamAssassinMaxScore	Float	Set the maximum score that the MIAS engine can reach. If the score is higher than this value, it's capped.
AddSubject	Boolean	Enables/disables adding text to Subject of spam message.
AddSubjectString	String	Specifies the text that is added to the Subject of spam messages when this feature is enabled.
OutgoingRules	Integer	Defines how the outgoing messages are processed (if at all). 0 = All outgoing messages are checked for being Spam by the Instant Anti-Spam engines. 1 = Messages sent out from the server will never be marked as Spam in this mode. 2 = This option ensures that outgoing messages that are identified by Instant Anti-Spam as Spam are rejected and not sent.
BypassLocalIPs	Boolean	All IP addresses located in the Relay IPs list, relay.dat file and also all authenticated sessions will be bypassed for the whole anti spam engine.
IgnoreMessagesLarger	Integer	Use this property to set a size limit of messages that should be processed by MIA. Messages larger than this limit (specified in kilobytes) are bypassed completely.
UseSpamFolder	Boolean	Setting this to 1 activates the Spam Folder Mode. With this feature enabled, Meral will place only genuine e-mails in the user's Inbox while separating e-mail identified as Spam into their Spam Folder.
DeleteSpamMailOlder	Integer	This defines a number of days after which a message is removed from the Spam folder.
SpamUseIMAP	Boolean	When set to 1, the Spam Folder will be used for IMAP accounts.
SpamBypassNonUsers	Boolean	By setting this to 1 you disable Anti-Spam for non-user entities created within mail server like mailing lists, static routes, notifications, ...
SpamIMAPFolder	String	Specifies the name of the IMAP folder that will be used for Spam e-mail collection.
SpamCustomLow	Float	When Anti-Spam user self control is enabled and user chooses a low spam detection level, this value is used as a score limit for genuine messages.
SpamCustomMedium	Float	When Anti-Spam user self control is enabled and user chooses a medium spam detection level, this value is used as a score limit for genuine messages.
SpamCustomHigh	Float	When Anti-Spam user self control is enabled and user chooses a high spam detection level, this value is used as a score limit for genuine messages.

SpamMaxThreads	Integer	Every message is processed in a unique thread. This specifies the maximum number of threads that your MIAS engine can use. When set to zero, there is no limit.
SpamUpdate	Boolean	This enables automatic the MIAS engine update.
SpamUpdateDays	Integer	Specifies the days when automatic MIAS update should be performed. Lower seven bits of this number specifies the days of the week starting by Sunday as the lowest one.
SpamUpdateTime	String	Defines the day time when MIAS update should occur. 24 hour time format is used here (e.g. 23:30).
SpamUpdateURL	String	URL that is used for searching for new MIAS updates. Basically this is supposed to be always set to http://www.icewarp.com/update/spam.xml
SpamUpdateProxy	String	Proxy server that is used for the MIAS update. Use standard format address:port#

Spam Assassin

CHAPTER 4

How It Works

Spam Assassin is the most important part of MIAS, with a bit of familiarity we can say it's the heart. It uses a wide variety of tests (including header and text analysis, Bayesian or network tests) to identify spam signatures. This makes it harder for spammers to identify one aspect which they can craft their messages to work around.

SA Scoring System

Hundreds of tests are applied on every message. The result of each test is a simple true/false answer that says whether the message failed in the test or not. Please, notice that failing in the test means meeting it's criteria (what's bad for most tests). Neither response determines a judgement, but a positive answer (test failed) can have consequences depending on the test intent. A simple scoring system is applied here. Each test has its own score associated with it, which is applied in the case the test failed. When all tests are finished, the resultant score is made of the sum of partial scores of all tests that failed (had positive answer). All these tests and their scores are written in *configuration files* (on page 18). We can discuss one sample entry:

```
score PENIS_ENLARGE 1.101 1.101 0.500 2.692
```

A line above defines a score for "PENIS_ENLARGE" test. Although the label is just a symbolic name, it's obvious that it checks messages for "penis enlargement" text presence. If this text is found in the message, the test has a positive answer (it's score is added to the total sum). We can see four (possibly different) scores defined, which is not necessary (read below). The score that is used depends on how Spam Assassin is being used. The first score is used when both Bayes and network tests are disabled (score set 0). The second score is used when Bayes is disabled, but network tests are enabled (score set 1). The third score is used when Bayes is enabled and network tests are disabled (score set 2). The fourth score is used when both, Bayes and network tests, are enabled (score set 3).

We can also find a line with only one score defined (see below). In that case this score is always used for that test. If this single number is zero, the test is disabled and therefore never applied.

```
score MANY_FROMS 0
score BODY_8BITS 1.500
score USER_IN_WHITELIST_TO -6.000
```

Note: The syntax described above is only for understanding and clarity, most likely you will never have to access and/or change these data items. However, if you understand this, you can find a use for a complete list of all SA tests: http://spamassassin.apache.org/tests_3_1_x.html

Positive numbers are used for the most part for test scores, but notice that even a negative score could be defined here. This score would also be added to the total sum when the requirements are met and it would lower the spam probability of the processed message. The most important point here is the resultant score of all tests is being applied. This score is later used in the final MIAS computations described in *How it works* (on page 3) chapter. It has no ranges, so it can be any real (positive or negative) number. Legitimate non-advertising messages will generally end with a score at or below zero.

There are two good reasons why you shouldn't change these scores in the configuration files:

1. These values are based on statistics made up from tons of messages all over the world, so they work pretty well in general.
2. MIAS update is performed time to time and would remove your changes (please, refer to this chapter: *configuration files* (on page 18)).

Reporting Function

Since there is a huge variety of tests, it would be hard to determine why any message was marked as spam even if you knew it's resultant score. That's why Spam Assassin offers a reporting function. It can make a list of all tests, in which a message had failed (got scored, even negatively). Three possible ways of reporting are available:

- § Adding SA headers
- § Generating report message
- § Extending original message

The second and third ways are very similar. Use them to generate a new message that is sent to the original recipient. This new message will contain Spam Assassin results in a readable aligned table. The difference between these methods is only that the second one generates a report message and the original is attached to this report, whereas the third method works reversely, thus adding the report message as an attachment to the original one. We can take a look at one such report:

```
Spam detection software, running on the system "crazyanimals.net", has
identified this incoming email as possible spam. The original message
has been attached to this so you can view it (if it isn't spam) or block
similar future email. If you have any questions, see
@@CONTACT_ADDRESS@@ for details.
```

Content preview:

Content analysis details: (6,28 points, 3,00 required)

```
pts rule name description
```

```
-----
4,1 SUBJ_VIAGRA Subject includes "viagra"
```

```
0,9 TO_MALFORMED To: has a malformed address
-0,7 BAYES_10 Bayesian spam probability is 10 to 20%
1,0 DISGUISE_VIAGRA Disguised word "viagra"
0,0 NO_RDNS2 Sending MTA has no reverse DNS
1,0 DRUGS_ERECTILE Refers to an erectile drug
```

As you see, this message is very useful if you want to get detailed information from Spam Assassin - a description is added to every test that failed.

Using SA headers is much less informative since it contains only the symbolic names of tests. It adds these names to a special message header thus preventing the user from detailed unwanted data. One such extended message header could look like this one (take note that it reflects the same spam message as the reporting message above):

```
X-Spam-Flag: YES
X-Spam-Status: Yes, hits=6,28 required=3,00
tests=SUBJ_VIAGRA,TO_MALFORMED,BAYES_10,DISGUISE_VIAGRA,NO_RDNS2,DRUGS_ERECTIL
Eversion=3.1
X-Spam-Level: *****
X-Spam-Checker-Version: SpamAssassin 3.1 (1.5) on crazyanimals.net
X-SpamReason: SpamAssassin=6,27
```

Actually, these headers are also added to the spam report message, so the report message is rather an extension of the basic Spam Assassin headers feature. You can see it contains only very brief information, but it can be sufficient in most cases and should not bother your users.

Additional Filters

There are some special advanced tests done by Spam Assassin that can be enabled or disabled separately in the Administration Console. For details, please check out the appropriate paper:

§ *SPF*

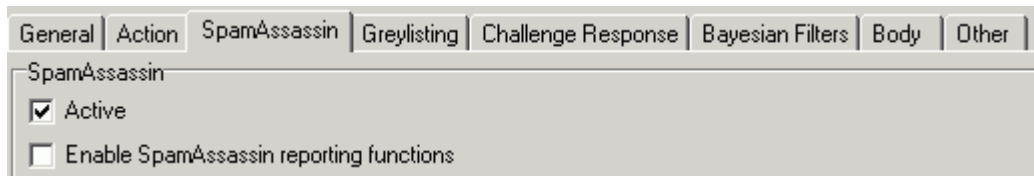
§ *SURBL*

§ *DomainKeys*

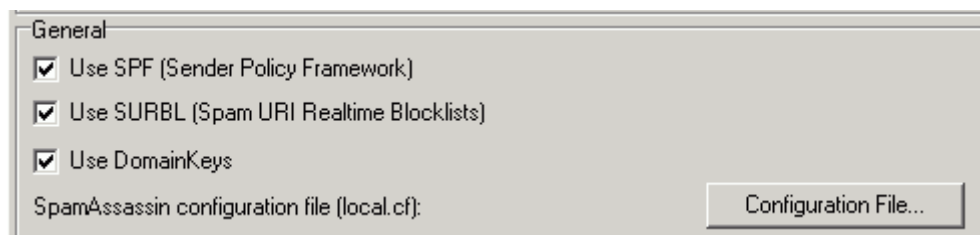
§ *Bayesian Filters* (on page 52)

CHAPTER 5

Administration



Field	Description
Active	Enables the SpamAssassin filters. It is recommended that this option be enabled.
Enable SpamAssassin reporting functions	SpamAssassin reporting function enables the reporting features of the SpamAssassin engine. Reports are written as x-headers to the original message, or a new report message is created and attached to the original e-mail. See the reporting section of this tab.



Field	Description
Include Bayesian probability in SpamAssassin score	This option allows the use of Bayesian filters as a part of the SpamAssassin scoring system. It increases the accuracy of the SpamAssassin engine.
Use SPF	Checking this option enables the Sender Policy Framework technology. You can find more information at http://spf.pobox.com/
Use SURBL	Checking this option enables the Spam URI Realtime Blocklist technology. You can find more information at http://www.surbl.org .

Use DomainKeys	Enables support of DomainKey technology for incoming email messages. If an incoming email from a domain which has a DNS DomainKey record that is not signed, the total "spam" score is increased. If an incoming email is not signed at all, the score is also increased (but less than in the first case)
Configuration file	The "Configuration file" button opens the local.cf, which defines SpamAssassin settings that override the original ones.

Reporting

- Report is added to headers and/or subject of the original message
- Generate report message (attach original message to report)
- Convert original message to text and attach to report message

These options are self-explanatory. Decide what kind of SpamAssassin reporting best fits your needs.

CHAPTER 6

Configuration Files

Spam Assassin was originally an open source project. It uses its own configuration files with a common syntax. We are not about to discuss here the entire possible configuration, just the most important parts for Merak. For further details please visit this project's home page (http://spamassassin.apache.org/full/3.0.x/dist/doc/Mail_SpamAssassin_Conf.html). We suppose here that you are already familiar with the SA scoring system.

All files that SA uses are stored in one folder: <MerakInstallPath>\spam\rules\. If you take a look at the folder, you'll find several files with the .cf extension. These files define score values for all SA tests. You can disable any filter here, change its actual score or change its header that is added to the SA report. However, any change made in any file but your local.cf file is discarded when MIAS update is performed. So if you are about to change any setting, make sure you are making the change in your local.cf file, so that your changes will be permanent. The important thing here is, that the local configuration overwrites the original one, so as an example, if you want to change the score of a Bayesian test with result between 70 and 80 percent, you have to copy the following line

```
score BAYES_70 0 0 2.142 2.255
```

from file 23_bayes.cf to local.cf and change it to reflect your needs, e.g.

```
score BAYES_70 0 0 4.284 4.51
```

This file also contains the basic SA setting that covers information about which main components (*Bayesian filter* (see "Bayesian Filters" on page 52), *SPF*, *Domain Keys*, ...) are about to be used. Some helpful properties are described below. You can also find a complete list of Spam Assassin tests: http://spamassassin.apache.org/tests_3_1_x.html.

Variable	Type	Description
required_hits	Float	Use this variable to define the score limit for genuine messages. Messages that reach this limit will be marked as spam.
use_bayes	Boolean	Set this to 1 to enable <i>Bayesian filter</i> (see "Bayesian Filters" on page 52).
use_spf	Boolean	Set this to 1 to enable <i>Sender Policy Framework</i> .
use_surbl	Boolean	Set this to 1 to enable Spam URI Realtime Blocklists.
use_domainkeys	Boolean	Set this to 1 to enable <i>Domain Keys</i> test.
subject_tag	String	If rewrite_subject property is set, this defines the text that is added as a prefix to the subject of spam message.
rewrite_subject	Boolean	Set this to 1 to enable marking spam messages by text added to subject. It's not recommended use this; use AddSubject variable in spam.dat file instead.

report_safe	Integer	Defines the type of SA reports. Three values are allowed here: 0 = Message headers are extended by brief spam report. 1 = New report message is generated and attached to the original message. 2 = A report message is generated and the original message is added to it as an attachment.
add_header	String	Use this variable to define custom headers that are added to some messages. The syntax of this parameter is as follows: <code>add_header target type name content</code> Target can be string „spam“, „genuine“ or „all“ and says which messages are extended by the header. Name is just the header name and can contain any string without spaces. The final header name is „X-Spam-name“. Content specifies the header value. It can be set to any string. You can use macros here, some useful are: <code>_YESNO_</code> is set to Yes or No, depending on the fact whether the message is spam. <code>_YESNOCAPS_</code> same as above, but uses capitalized characters. <code>_HITS_</code> prints out the resultant spam score. <code>_REQD_</code> contains the value of required_hits variable. <code>_TESTS_</code> lists all tests that failed. <code>_VERSION_</code> contains the SA version. <code>_SUBVERSION_</code> contains the SA minor version. <code>_STARS(*)_</code> displays the spam probability of the message by a line of asterisks (one for every point of spam score reached). You can even set any other character. <code>_HOSTNAME_</code> contains the server hostname.

There are also some properties in spam.dat file that are needed for Spam Assassin configuration. Follows a brief description:

MIAS Properties in Detail

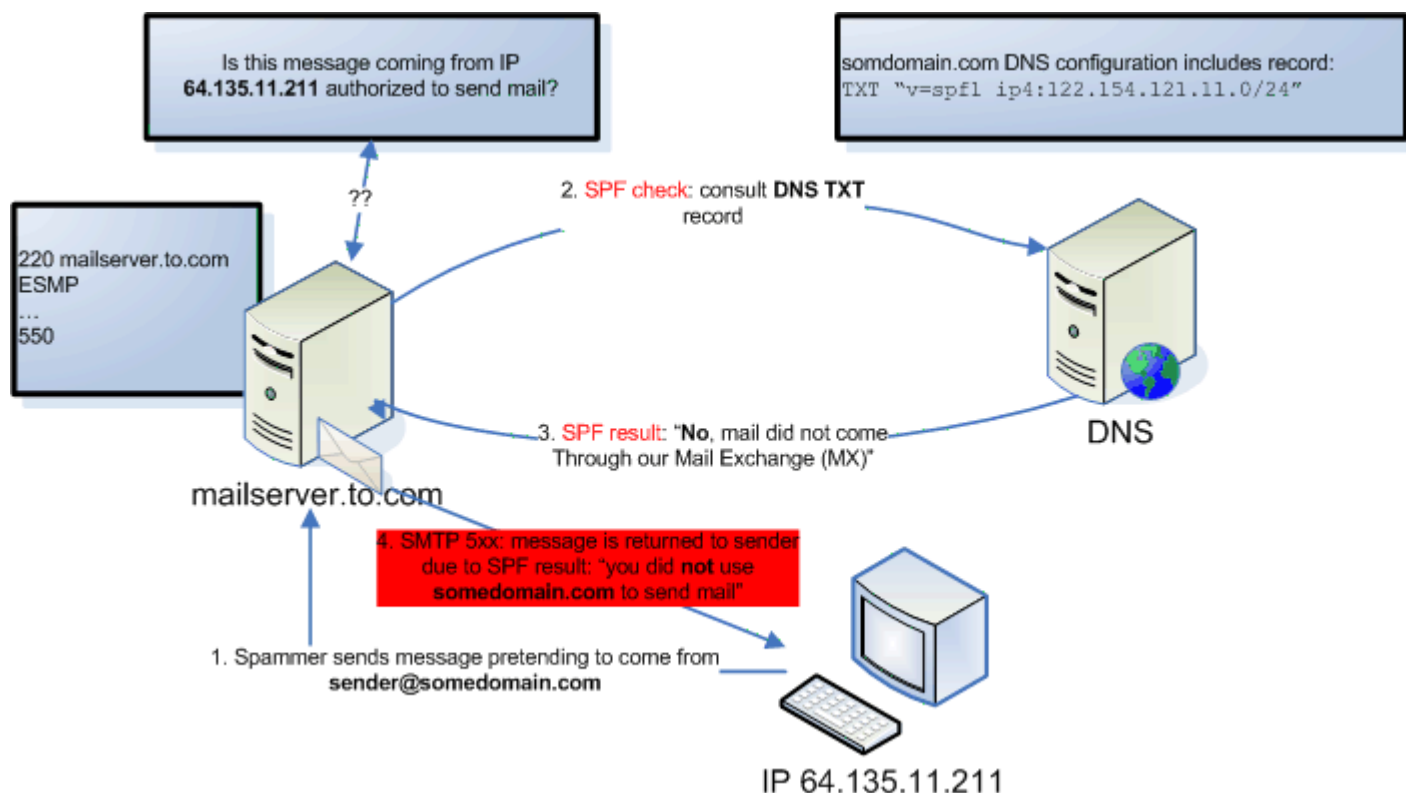
Here we will list and briefly describe some properties that can be used in the spam.dat file.

Variable	Type	Description
SpamAssassinEnabled	Boolean	Set this to 1 to enable Spam Assassin.
SpamAssassinMarked	Boolean	When this is set to 1, Spam Assassin is applied on messages that were already marked as spam.
SpamAssassinRulesPath	String	Set this if you want to define a custom path to Spam Assassin configuration files. Leave it blank to let SA use the default one.
SpamAssassinReporting	Boolean	Setting this to 1 enables Spam Assassin reporting function.

Sender Policy Framework and Sender Rewriting Scheme

The SMTP server normally permits anyone to send an email message from an email address that could be possessed by someone completely different than the actual sender of the message. This helps spammers and senders of unsolicited mails to conceal their true identity and send messages that look like they were sent from someone the receiver would usually trust. For example, a spam message offering some annoying crap can arrive from the email address of your mother... This is considered to be a security hole in SMTP and SPF is one of the most effective techniques at preventing such behaviour.

SPF technology allows the domain owner to add an additional DNS record for the domain owned, stating machines that are authorized to send email messages from this domain. Any receiver machine that is SPF compliant will find suspicious all mails claiming to come from a domain, that fails the authorized location analysis that the domain has defined in their DNS.

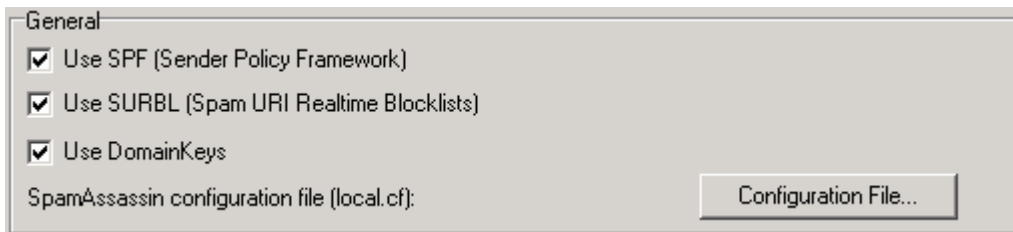


Merak Mail Server from version 8.2.0 offers the SPF technology as a part of the Merak Instant AntiSpam module. Merak Instant AntiSpam - one of the most complex and powerful AntiSpam packages in the world - after implementing SPF has become even more effective in AntiSpam fighting.

One of the problems that comes with SPF is SMTP forwarding (where an email server forwards email to someone else without changing the "mail from" address in the SMTP session) is easily solved by implementing SRS (Sender Rewriting Scheme). SRS forces the rewriting of the "mail from" address by the forwarding agent. Merak Mail Server with its Merak Instant Anti Spam module comes with both SPF and SRS (among dozens of others) and thus is one of the most secured email server solutions in the world.

Administration

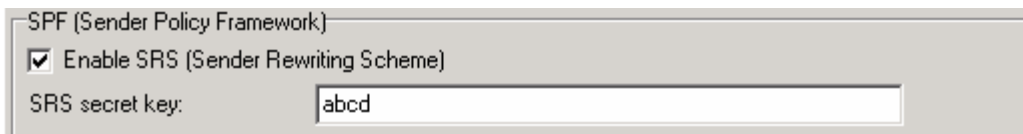
Sender Policy Framework is implemented in Spam Assassin so it can be enabled or disabled by a checkbox under the Spam Assassin tab:



If you want to enable it by editing its configuration file, open Spam Assassin local configuration file (spam\rules\local.cf) and set the use_spf variable to 1.

As a part of Spam Assassin, SPF doesn't have its own bypass file.

The Sender Rewriting Scheme is implemented by Merak itself however, so it has to be enabled in the Protection section of the Mail Service Security setting:

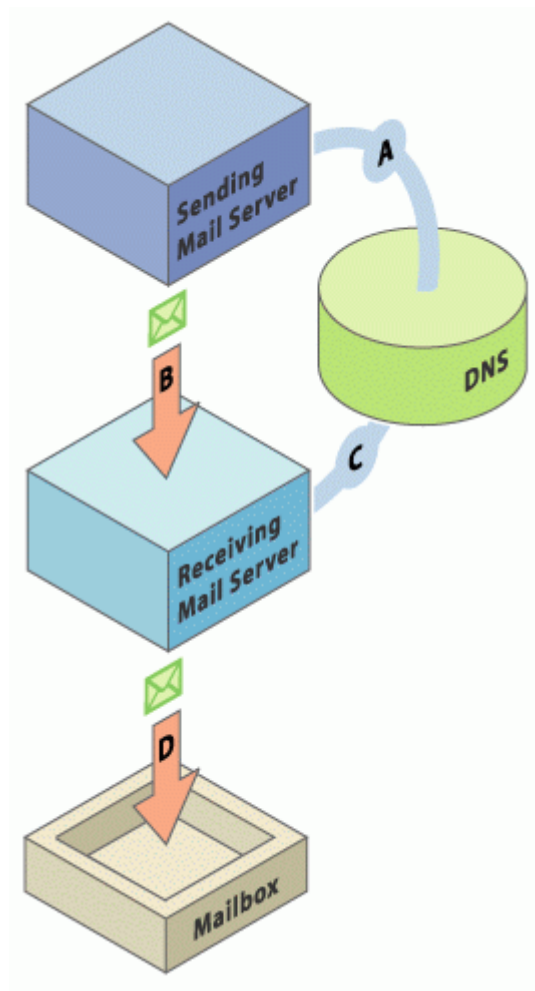


If you want to set up a SPF record for your domain, you can find a useful SPF wizard available on this page: <http://www.openspf.org/>.

There are several site on the Internet that allow you to check SPF records of some particular domain. If you need one, try this one for example: <http://www.skylist.net/resources/authentication.php>.

DomainKeys

Under DomainKeys, a domain owner generates one or more private/public key-pairs that will be used to sign messages originating from that domain. The domain owner places the public-key in his domain namespace (i.e., in a DNS record associated with that domain), and makes the private-key available to the outbound email system. When an email is submitted by an authorized user of that domain, the email system uses the private-key to digitally sign the email associated with the sending domain. The signature is added as a header to the email, and the message is transferred to its recipients in the usual way.



[source: <http://antispam.yahoo.com/domainkeys>]

How it Works - Sending Servers

There are two steps to signing an email with DomainKeys:

1. Set up: The domain owner (typically the team running the email systems within a company or service provider) generates a public/private key pair to use for signing all outgoing messages (multiple key pairs are allowed). The public key is published in DNS, and the private key is made available to their DomainKey-enabled outbound email servers. This is step "A" in the diagram to the right.

2. Signing: When each email is sent by an authorized end-user within the domain, the DomainKey-enabled email system automatically uses the stored private key to generate a digital signature of the message. This signature is then pre-pended as a header to the email, and the email is sent on to the target recipient's mail server. This is step "B" in the diagram to the right.

How it Works - Receiving Servers

There are three steps to verifying a signed email:

1. Preparing: The DomainKeys-enabled receiving email system extracts the signature and claimed From: domain from the email headers and fetches the public key from DNS for the claimed From: domain. This is step "C" in the diagram to the right.

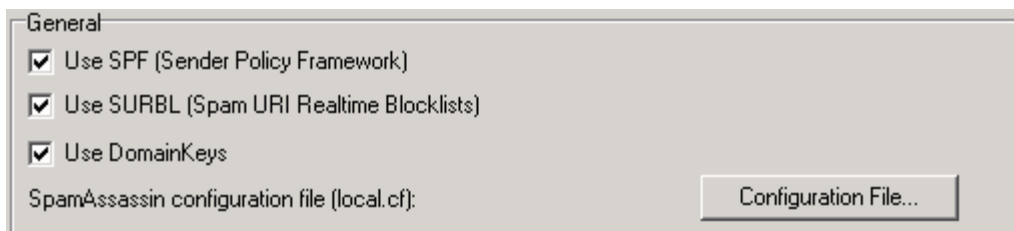
2. Verifying: The public key from DNS is then used by the receiving mail system to verify that the signature was generated by the matching private key. This proves that the email was truly sent by, and with the permission of, the claimed sending From: domain and that its headers and content weren't altered during transfer.

3. Delivering: The receiving email system applies local policies based on the results of the signature test. If the domain is verified and other anti-spam tests don't catch it, the email can be delivered to the user's inbox. If the signature fails to verify, or there isn't one, the email can be dropped, flagged, or quarantined. This is step "D" in the diagram on the right.

[source: <http://antispam.yahoo.com/domainkeys>]

Administration

The only thing you have to do is to activate the DomainKeys functionality in Anti-Spam configuration, Spam Assassin tab:

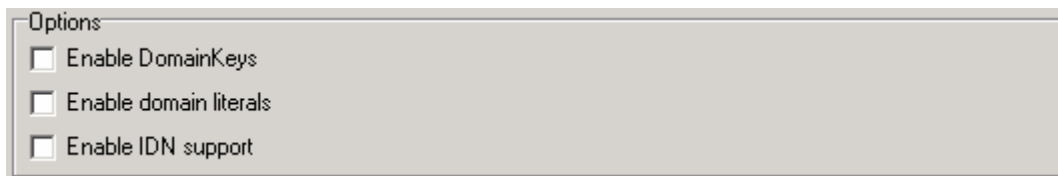


If you want to enable it by editing the configuration file, open Spam Assassin local configuration file (spam\rules\local.cf) and set the use_domainkeys variable to 1.

There's no special bypass file for Domain Keys - it uses the one used by MIAS.

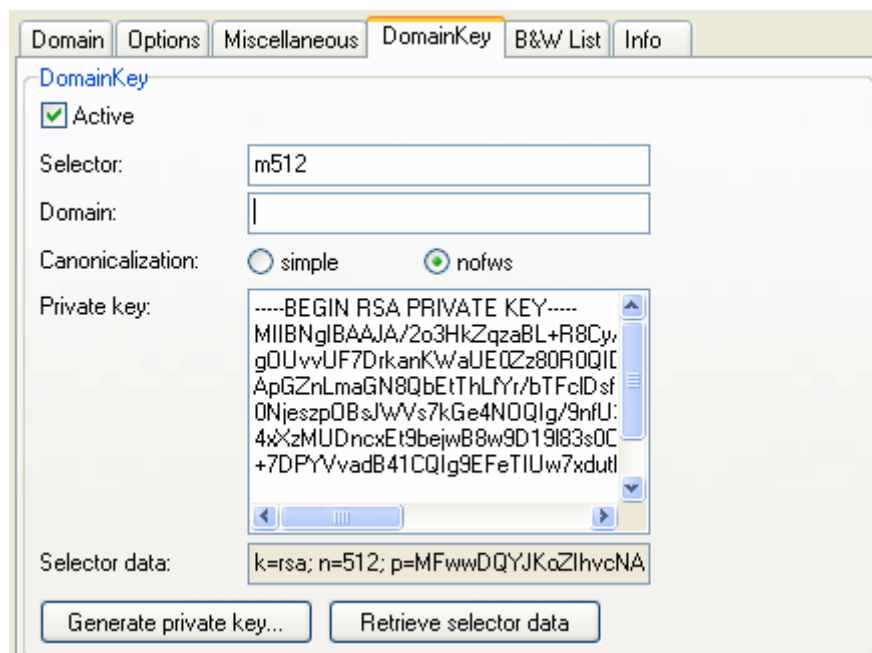
However, you will probably want more than this. You may want to configure some of your domains to use DomainKeys also. In that case, follow these steps:

1. Enable the DomainKeys for your domains globally in Global Settings section under Domains tab:



2. Configure some particular domain to use DomainKeys. You can do this in the DomainKey tab of that domain configuration section. For simplicity you can avoid unnecessary steps and thus you only have to

- a) Check on the Active check box
- b) Fill out the Selector field - create some text identifier you will later use on your DNS
- c) Click on Generate private key button



3. Configure your DNS server. If you click on Retrieve selector data button as shown in the image above, the Selector data field will be filled with a value that you will use on your DNS. Just create one TXT record in the form [selector]_domainkey.[yourdomain.tld] and set it to Selector data value obtained from Merak.

There are several pages over the Internet that allows you to check DomainKey records of some particular domain. If you need one, try this for example: <http://www.skylist.net/resources/authentication.php>.

CHAPTER 7

Greylisting

In This Chapter

How It Works	36
Administration	38
Configuration Files	40

How It Works

Greylisting is a fancy method of spam controlling. It is based on the fact that spammers and unsolicited email senders do not use machines compliant with internet standards. Greylisting prevents spam by responding with a temporary SMTP error after the first attempt for message delivery. 99% of spam and viruses are sent from mail bombers (automated mail sending programs) which do not ever try to deliver the mail again, so these are blocked for good. Normal mailservers retry after a temporary error a bit later and Greylisting allows the message through.

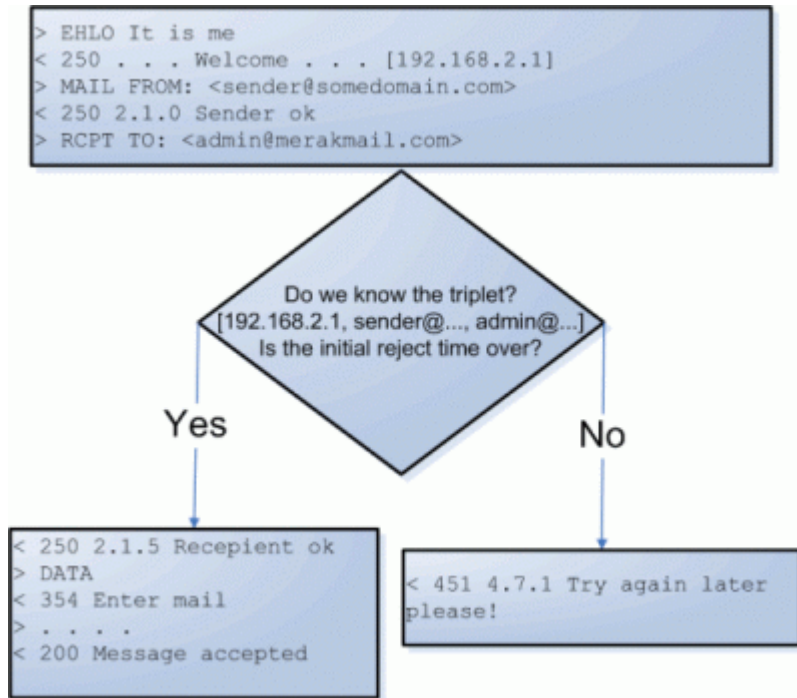
The flash characteristics of greylisting would probably be an exclamation: "How primitive! Although, so efficient!" Greylisting is identifying and blocking spam with unbelievable efficiency at a rate of about 80% using an entirely trivial method: temporarily suspend mail delivery. However if the mail was sent by a well-mannered mail server, the message will appear in your inbox.

The cornerstone of greylisting is a simple idea. A well-mannered and internet-standard compliant mail server is trying to deliver the message even though it encounters a temporary rejection. The mail server (or MTA, Mail Transfer Agent) would, after being rejected, put the message into a queue and would try the delivery after a time period again. On the other hand, the overwhelming majority of spam is sent by mail bombers (specialized mail sending programs), trying to deliver vast amounts of messages to a huge number of recipients in a short amount of time. These automated mail bombers are too eager to send and deliver and do not ever bear to wait for responses or ever bear to retry the unsuccessful delivery of their spam messages.

The Greylisting is implemented on the recipient side - in Merak Mail Server. Merak Mail Servers record three pieces of information at the moment any e-mail is being received:

- § The IP address of the machine sending the e-mail.
- § The e-mail address to which the e-mail is being delivered.
- § Actual time.

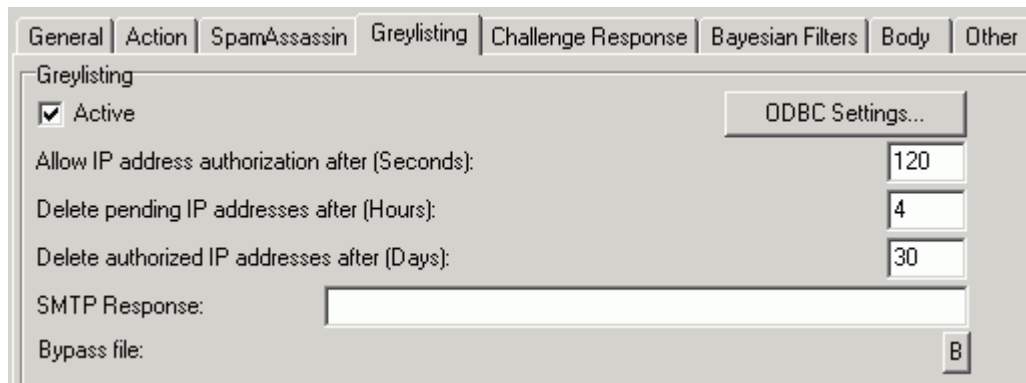
In the moment Merak captures this data, it will look into its database and compare the information with the data in the internal database. If the first two pieces of information matches with some entry in the database, the actual time is compared with a sum of the time stored in the database along with an acceptable time limit (you can configure this). If the actual time is bigger, the entry in the database is set to be authorized. If this pair (sending IP x recipient address) was already authorized before, then no check is made and Merak will deliver the message normally into the user's mailbox.



In compliance with the Internet standards specification, when a mail server receives a temporary "4xx" error, it will have to queue the message and retry to deliver it at later time. For genuine email messages and genuine mail servers, this process is normal and standard. Correctly configured mail servers will redeliver their messages and thus greylisting will not represent a delivery challenge. In most cases, the first retry is made within 2 minutes (but even 5 minutes is still an acceptable delay for the first message). On the other hand, the applications used by spammers do not redeliver these messages because it would decrease the total number of messages they would be able to send. If Greylisting would become a standard method that is deployed, the number of incoming spam messages would be reduced.

One of the advantages is that the mail message is rejected at the moment of its arrival and thus the message itself is never actually received by your mail server. This is helping you to save your connectivity and internet resources. Our tests with a 5 minute delay - which is bearable - have proven that as high as 80% of all spam was destroyed by Greylisting alone. If you are not convinced, see the statistical data at other websites dealing with greylisting, for example greylisting.org.

Administration



Please note, that for greylisting, these local bypasses are important:

- § Bypass trusted IPs,
- § Exclude outgoing messages from spam scanning,
- § Local-Local bypass filter.
- § Greylisting bypass file (greylist.dat)

If these are not applied, the users will get error 4.5.1 in their mail clients and are allowed to send the message after x seconds.

Field	Description
Active	Enables/Disables the Greylisting system. All data related to the Greylisting system are stored in the database via ODBC.
ODBC Settings	<p>The default ODBC source is created during installation automatically. It uses the "/merak/spam/challenge.mdb" Microsoft Access database file.</p> <p>By clicking on the "ODBC Settings" button you can specify another ODBC source or even a backup. The structure of the table for the new ODBC source must be exactly the same as in the default database.</p> <p>The challenge.mdb file is shared between Greylisting and Challenge Response. Once created in one (CR or Greylisting) you do not have to create it again in the other one. Greylisting creates table 'IPs'.</p>
Allow IP address authorization after (Seconds)	This is the temporary rejection time of the incoming session. If the originating server is attempting to retry the delivery during this time period it is rejected.
Delete pending IP addresses after (Hour)	If the originating server will not retry to deliver the mail within this time period, the server IP is deleted from the database. The time period is in hours.

Delete authorized IP addresses after (Days)	<p>If the server IP is authorized it is deleted after here specified number of days. This is security option ensuring that the authorized servers have to go through Greylisting process after several days again.</p> <p>Note: If you want to keep authorized servers list forever, set the value to 0</p>
Bypass file (greylist.dat)	Greylisting bypass file.

You will probably want to check out which servers are accepted or greylisted from time to time. To be able to do this, you have to know the database structure. The database used for greylisting is shared with the Challenge Response mechanism and contains only two tables - IPs and Senders. Greylisting uses the IPs table, which consist of these columns:

- § ipAddress contains the sender's IP.
- § ipEmail stores the recipient's email, this field together with ipAddress makes up the key attribute.
- § ipAuthorized field can contain only two values - 1 or 0. Zero signifies a greylisted sender whereas 1 is used for authorized senders.
- § ipCreationDate stores the record creation date. This is used for compute authorization lifetime.
- § ipCreationTime specifies the exact time of the first message delivery attempt. This is used to compute a pending items lifetime.

Configuration Files

Greylisting configuration is stored in the main MIAS configuration file spam.dat. The list of relevant properties follows:

Variable	Type	Description
SpamChallengeConnection	String	ODBC connection string for shared DSN used by Challenge Response and Greylisting. Used syntax is: DSN;login;password DSN2;login2;password2 When the first connection fails, the second one is used. Only the DSN parameter mandatory; login, password and even the backup connection are optional.
SpamGLActive	Boolean	Setting this to 1 activate the Greylisting feature.
SpamGLAllow	Integer	This property is used for Greylisting. It specifies the temporary rejection time (in seconds) of the incoming session. If the originating server is attempting to retry the delivery during this time period it is rejected.
SpamGLPending	Integer	If Greylisting is enabled and the originating server (that was firstly refused) will not retry to deliver the mail within this time period, the server IP is deleted from the database. The time period is in hours.
SpamGLAuthorized	Integer	If the server IP is authorized it is deleted after here specified number of days. This is security option ensuring that the authorized servers have to go through Greylisting process after several days again. If you want to keep authorized server list forever, set the value to 0.

For better adjustability a specific bypass file (config\greylist.dat) is available. It's applied only for greylisting.

CHAPTER 8

Challenge Response

In This Chapter

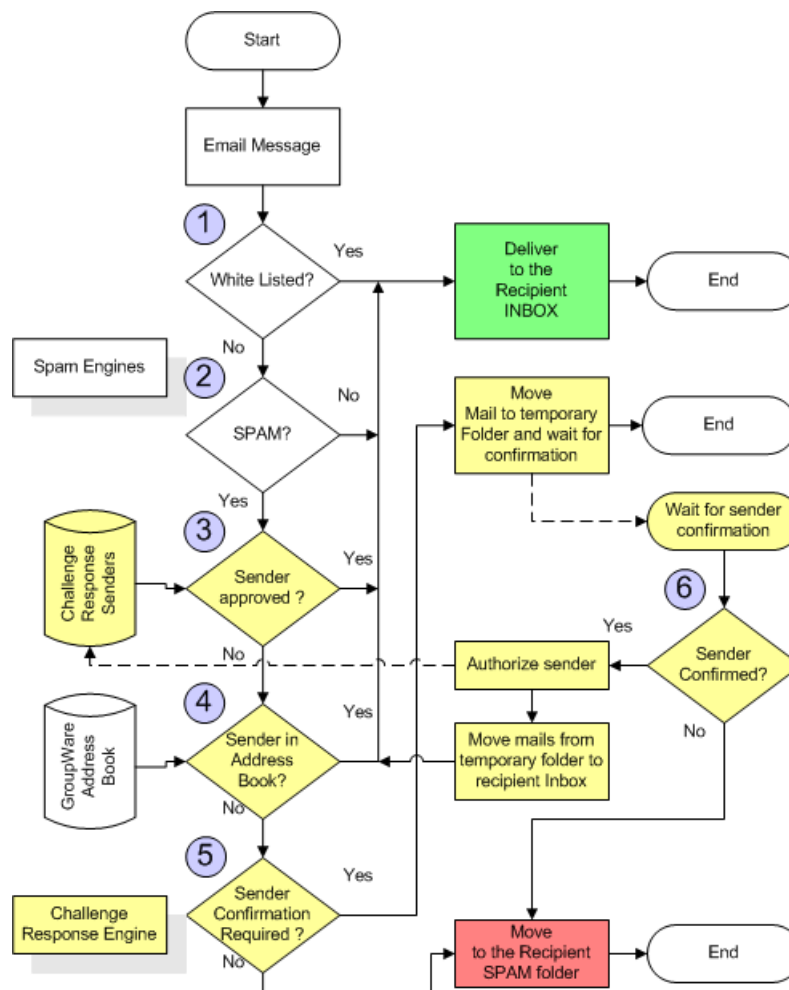
How It Works	42
Administration	46
Configuration File.....	51

How It Works

Challenge/Response is a system that requires the sender of an email to verify that he/she actually sent the email. Re-confirmation must be provided manually for Challenge/Response to work.

If the e-mail was sent by a mass-mailing system, there is usually no human at the email address that is used in the "From" header, thus re-confirmation cannot be provided.

The Challenge/Response system is a critical component of the full Anti-Spam solution. The yellow components below are the full Instant Anti-Spam data diagram.



In the most typical situation, messages arrive at the Challenge/Response system after they have already passed all "white listing" possibilities as described in the Black & White Listing Techniques and are already marked as Spam.

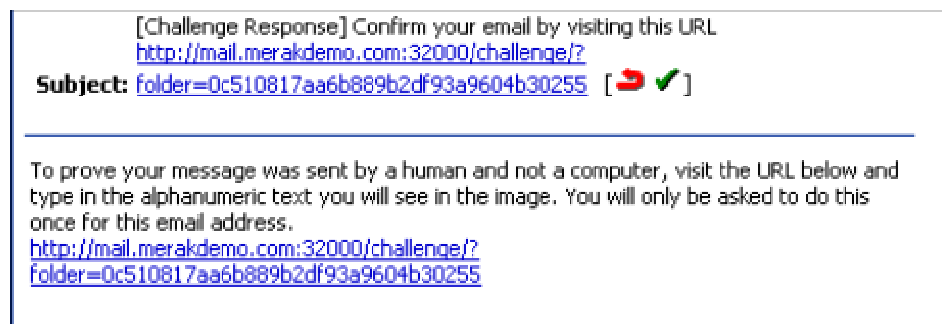
- § When the email is received by the server, it is not delivered to the recipient, but stored in a temporary folder. If more messages are sent from the same sender then all messages are stored in the same folder. Such messages are marked as "pending message(s)". If the pending message is not authorized within the specified number of days - it is automatically deleted.
- § The Server will generate the request for confirmation, which will be delivered to the sender of the e-mail. It uses the sender from the SMTP protocol, which can be different from the "Mail From:" displayed in the message.
- § The Sender (if they exist) will receive the request for confirmation and must confirm it. The confirmation requires visiting a special web site and entering some characters in a text field. It prevents usage of automated confirmation systems.
- § The Server will receive the confirmation from the sender and will deliver the e-mail(s) to the recipient. The sender is also entered to the "approved senders list" so confirmation will not be requested the next time.

Emails with blank Mail From (it looks like MAIL FROM: <> in SMTP session) are bypassed by the Challenge Response engine. To handle such messages you should use Content Filters or Black & White Lists.

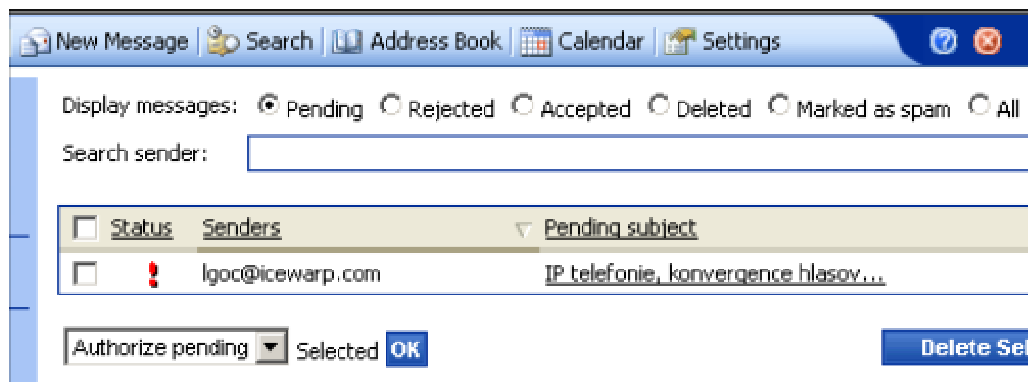
Tip: If you enable the Challenge Response engine, but disable sending of the challenge message, you will create a scenario like the CR would not be enabled, but if you respond to a message sent to you from a legitimate person, the sender will be automatically whitelisted and his message won't be marked as spam never again.

Screenshot Examples:

Request for confirmation sent by the mail server to the sender



Sender waiting for authorization - pending in the database



The URL of the page with sender confirmation request

To prove your message was sent by a human and not a computer, type in the alphanumeric text you see in the image below and click OK. You will only be asked to do this once for this email address.

QC46T-QSVPD

Thank you for your cooperation!

Why am I doing this?

Unsolicited commercial email is computer-generated and cannot respond to the command above. By using this permission-based email system, I am restricting my inbound email to senders who authenticate, providing they are real humans who wish to communicate with me via email.

Thank you for helping me banish spam!

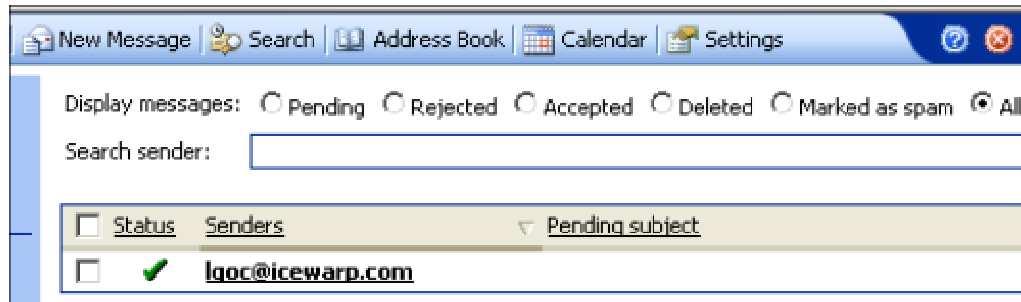
If the sender enters the code properly they are automatically authorized

To prove your message was sent by a human and not a computer, type in the alphanumeric text you see in the image below and click OK. You will only be asked to do this once for this email address.

The word you specified is correct. Your email address has been authorized.

Thank you for your cooperation!

Sender is added to the Challenge Response as authorized.



Depending on the setup of the Challenge response system, the sender can be authorized for just one recipient, or for all recipients on the server.

Administration

The Challenge Response system requires senders from external domains to authorize their email address for future communication with your mail server. External senders will only be challenged once and will need to authenticate their message. Alternatively, local users of your mail server can authorize/accept the message in Webmail - Settings - Challenge Response.

Email messages that are From: any local domain in your mail server will by-pass the Challenge Response system, this includes mail from Internet users or spammers spoofing the From: address. In this case it's recommended that you enable "Reject if originators domain is local and not authorized" located in the Security - Anti Relaying tab.

Field	Description
Active	Enables/Disables the Challenge/Response system. All data related to the Challenge/Response system is stored in the database via ODBC.
ODBC Settings	<p>The default ODBC source is created during installation automatically. It uses the "/merak/spam/challenge.mdb" Microsoft Access database file.</p> <p>By clicking on the "ODBC Settings" button you can specify another ODBC source or even a backup. The structure of the table for the new ODBC source must be exactly the same as in the default database.</p> <p>The challenge.mdb file is shared between Greylisting and Challenge Response. Once created in one (CR or Greylisting) you do not have to create it again in the other one. Challenge Response creates table 'Senders'.</p>

Automatically add outbound message recipients to whitelist	<p>This check box allows for the automated building of the "authorized senders" list without the "confirmation pending" mechanism.</p> <p>If checked, any recipient to which a user sends a message to will be automatically added to the "authorized" list. When the recipient responds to the e-mail, they are already an authorized sender and the message will be automatically delivered to the user.</p> <p>It is strongly recommended that this option remain checked</p>
Apply Challenge Response if score is between	<p>"Apply Challenge Response" means that the server will send a request for authorization automatically to the sender.</p> <p>With this option, you can specify when to send a request based on the score result.</p> <p>By default, mail is recognized as Spam if the score is 3.00</p>
Apply Challenge Response to messages marked as spam.	<p>With this option, all messages that are recognized as Spam require sender authorization.</p> <p>By checking this option you will decrease false positives (genuine messages in the Spam folders)</p>
Apply Challenge Response to messages not marked as spam.	<p>With this option, all genuine messages require sender authorization.</p> <p>This option you will decrease false negatives (Spams in the inbox)</p> <p>Even if both options are checked, regular senders are not processed as they are usually already "approved" or white listed. However it is important to keep in mind that it can increase traffic to your server.</p>
Days before a pending message is automatically deleted	<p>How long the message waits for confirmation. 0 means it waits forever.</p>
Only one challenge email to sender / Challenge e-mail to sender for each user	<p>Only one challenge email to sender</p> <p>Only one list of allowed senders will be used for whole server. So once sender is approved he can send to any user on the server.</p> <p>Acceptance is based on the GroupWare address book's use of the PUBLIC address book.</p> <p>This mode is suitable for businesses that have deployed Merak Mail Server for their own exclusive use.</p> <p>Challenge e-mail to sender for each user</p> <p>If checked, each user on the mail server will have separate list of approved senders.</p> <p>Acceptance is based on the GroupWare address book's use of the PRIVATE address book of each user on the server.</p> <p>This mode is useful for Internet Service providers.</p>
Processing Mode	<p>Processing mode allows for the enabling/disabling of the Challenge/Response system for individual users or domains as already described in Server - Domain - User Processing.</p>

The confirmation request that is delivered to the sender by Merak Mail Server contains a URL that must be accessed in order to process the sender's confirmation.

This URL points to a page that is served by Merak's Integrated Webserver that is installed by default with Merak. This same engine is used by the Web-based Administration and by Web Mail.

Customization

Confirmation web site URL:

Challenge email sender:

Challenge email from:

Challenge email customization:

Field	Description
Confirmation web site URL	<p>The default confirmation web site URL matches the settings of the Merak System. It uses the same port and hostname as is used for Merak integrated web mail access.</p> <p>Multi-domain systems should instead use the server hostname with the system variable %%Recipient_Domain%%.</p> <p>http://%%Recipient_Domain%%:32000/challenge/</p> <p>Each confirmation URL generated by the mail server will match the recipient domain name. Set your DNS server correspondingly.</p>
Challenge email sender	<p>The specified sender will be used in the SMTP protocol.</p> <p>It is recommended that the default setting remain (empty), as some mail server autoresponders automatically respond to the senders address.</p> <p>If you were to enter an address here, it will simply increase non-essential traffic to your server.</p>
Challenge email from	<p>This value is used in the challenge response request confirmation e-mail.</p> <p>A value should be entered here as e-mails containing an empty "sender" and "from" will be rejected by many mail servers. However, it is not necessary to enter a real e-mail address there.</p>
Challenge email customization.	<p>This feature allows you to modify the text of the email that the challenge/response system delivers to the sender of the rejected email.</p> <p>The new confirmation mail text is stored in /merak/spam/challenge.txt</p> <p>The Confirmation URL is passed to this file by using the variable "%s"</p> <p>Syntax of the file challenge.txt is very simple:</p> <ol style="list-style-type: none"> 1 line 1 - Subject of the mail 2 line 2 - empty (CRLF, CRLF) 3 next lines - body of the mail

Example:

The following confirmation request message has been generated by the mail server in response to the sender user@merakdemo.com who sent a message to the user xxx@webmail.domaina.com.

The Challenge Response URL was defined as: http://%%Recipient_Domain%:32000/challenge/

From:

To: <user@merakdemo.com>

Received: from webmail.domaina.com

by mail.merakdemo.com (Merak 7.2.3) with SMTP id DEMO

for <user@merakdemo.com>; Sun, 07 Mar 2004 01:48:16 +0100

Date: Sun, 07 Mar 2004 01:48:16 +0100

From: Challenge Response <info@merakdemo.com>

To: xxx@webmail.domaina.com

Message-Id: <812060168@mail.merakdemo.com>

Subject: [Challenge Response] Confirm your email by visiting this URL
<http://mail.merakdemo.com:32000/challenge/?folder=c42c1a770e2d6d07ff358b2c22d7cf71>

To prove your message was sent by a human and not a computer, visit the URL below and type in the alphanumeric text you will see in the image. You will only be asked to do this once for this email address.

<http://webmail.domaina.com:32000/challenge/?folder=c42c1a770e2d6d07ff358b2c22d7cf71>

If you want to have a control over the challenged senders, you can do this using Challenge Queue list which you can find in Status section. There you can filter all items so that only pending or only rejected messages are displayed etc. See the picture below. You can use this list also for simple deletion of any item.

Challenge Queue							
Status:	<input type="text" value="All"/>	Sender:	<input type="text"/>	Owner:	<input type="text"/>	Max Items:	<input type="text" value="1000"/>
Sender	Status	Date	Owner				
tempsoft@seznam.cz	Pending	2006-01-13	c@tempsoft.crazyanimals.net				
roman@support.icewarp.com	Accept	2005-12-18	roman@tempsoft.crazyanimals.net				
tempicek@gmail.com	Accept	2005-12-16	f@tempsoft.crazyanimals.net				
tempicek@gmail.com	Accept	2005-12-16	g@tempsoft.crazyanimals.net				
gotcha@doc.icewarp.com	Accept	2005-12-16	a@tempsoft.crazyanimals.net				
tempicek@gmail.com	Reject	2005-12-15	b@tempsoft.crazyanimals.net				
tempicek@gmail.com	Accept	2005-12-14	e@tempsoft.crazyanimals.net				
tempsoft@seznam.cz	Reject	2005-11-23	b@tempsoft.crazyanimals.net				
whoever@aol.com	Accept	2005-11-18	roman@tempsoft.crazyanimals.net				
tempsoft@seznam.cz	Accept	2005-11-16	d@tempsoft.crazyanimals.net				
tempsoft@seznam.cz	Spam	2005-11-11	g@tempsoft.crazyanimals.net				
tempsoft@seznam.cz	Reject	2005-11-11	f@tempsoft.crazyanimals.net				

If you are not able to use the Merak configuration console, you will surely welcome the database structure used for Challenge Response records. The database used for this is shared with the Greylisting engine and contains only two tables - IPs and Senders. CR module uses the Senders table only. Here are the important columns you should know:

- § SndEmail column contains the sender address.
- § SndAuthorized specifies the authorization status of this record. When this is set to 1, the sender is authorized to send a message to an appropriate recipient. If it is set to other value, he is not allowed, where 0 stands for 'spam', 2 stands for 'pending', 3 for 'rejected' and 4 means 'deleted by recipient'.
- § SndCreateOn field contains message delivery date. This is used for compute pending message lifetime.
- § SndFolder specifies the generated string assigned to the sender.
- § SndOwner column is used to store message recipient.

Configuration File

Challenge Response uses the main MIAS configuration file (spam\spam.dat) and bypass file (spam\spambypass.dat). A list of useful properties follows:

Variable	Type	Description
SpamChallenge	Boolean	Says whether the Challenge Response engine is enabled. (To enable it, set this variable to 1.)
SpamChallengeAddOutgoing	Boolean	Enabling this allows for the automated building of the "authorized senders" list without the "confirmation pending" mechanism. If set to 1, any recipient which a user sends a message to will be automatically added to the "authorized" list. When the recipient responds to the e-mail, they are already an authorized sender and the message will be automatically delivered to the user.
SpamChallengeMarked	Boolean	Setting this to 1 configures the Challenge Response mechanism to be used for messages marked as spam.
SpamChallengeUnMarked	Boolean	Setting this to 1 configures the Challenge Response mechanism to be used for messages not marked as spam.
SpamChallengeExpires	Integer	Use this variable to define count of days after which an unhandled (unconfirmed the sender, not rejected nor accepted by the recipient) challenged message will be considered expired and thus moved from CR database to user's mailbox while getting spam flag.
SpamChallengePerc	Boolean	Setting this to 1 enables applying Challenge Response mechanism on messages that are in specified Bayesian spam probability range. (see below)
SpamChallengeSpam	Float	This defines maximal value of Bayesian spam probability for which a message is challenged.
SpamChallengeGenuine	Float	This defines minimal value of Bayesian spam probability for which a message is challenged.
SpamChallengeSeparateUsers	Boolean	When this is set to 1, each user on the mail server will have separate list of approved senders. Acceptance is based on the GroupWare address book's use of the PRIVATE address book of each user on the server. When set to 0, only one list of allowed senders will be used for whole server. So once sender is approved he can send to any user on the server. Acceptance is based on the GroupWare address book's use of the PUBLIC address book.
SpamChallengeSAPerc	Boolean	When this is set to 1, Challenge Response mechanism is applied on messages that received specified score from Spam Assassin.
SpamChallengeSASpam	Float	This defines maximal score obtained from Spam Assassin for which the Challenge Response mechanism is applied.
SpamChallengeSAGenuine	Float	This defines minimal score obtained from Spam Assassin for which the Challenge Response mechanism is applied.
SpamChallengeSendOnce	Boolean	Set this to 1 to allow only one Challenge Response email per sender.
SpamChallengeURL	String	Defines URL that is sent to challenged sender for confirmation of his message.

SpamChallengeConnection	String	ODBC connection string for shared DSN used by Challenge Response and Greylisting. Used syntax is: DSN;login;password DSN2;login2;password2 When the first connection fails, the second one is used. Only the DSN parameter mandatory; login, password and even the backup connection are optional.
SpamChallengeEmailFrom	String	The specified sender will be used in the SMTP session when sending Challenge Response email.
SpamChallengeMailFrom	String	String used for the From header of Challenge Response email.

Bayesian Filters

CHAPTER 9

How It Works

Bayesian filters are dynamic statistical filters.

The statistical occurrence of words in an e-mail are compared with the "Reference Base" which results in a "probability" that that e-mail is or is not Spam. The result (percentage value) is then substituted by a score value that is finally added to the Spam Assassin score total. For this conversion a simple table is used that can be configured in the Spam Assassin configuration file. Lets see the following example spam message that the Bayesian filter recognizes without a doubt as 99% spam (the high occurrence of words common in spam messages convicts this message very easily):

From: Morgan <lowlands@linares.com>
Subject: Viagra, Xanax, Cialis, more...
Date: Tue, 16 Aug 2005 15:55:52 -0400

Buy Cialis Online! Get Control Of Your Life Again!
<http://nfp.p4aib8p0mzpftgp.dmtintymk.com>
Think like a man of action, act like a man of thought.
You can't build a reputation on what you are going to do.
Furious activity is no substitute for understanding.

How exactly this works? Bayesian email filters take advantage of Bayes' theorem. Bayes' theorem, in the context of spam, says that the probability that an email is spam, given that it has certain words in it, is equal to the probability of finding those certain words in spam email, times the probability that any email is spam, divided by the probability of finding those words in any email. This can be expressed by an equation

$$P(\text{spam}|\text{words}) = \frac{P(\text{words}|\text{spam})P(\text{spam})}{P(\text{words})}$$

Bayesian Filters are extremely powerful and accurate, however they will only function properly if the following conditions are filled:

- § The Spam Reference Base needs to be "taught" to recognize Spam e-mails. This process is called Indexing.
- § The e-mail messages that are Spam MUST contain some "content" that can be recognized.
- § The filter will work for any language that contains word separations (most languages except Chinese etc.)

Merak's Spam Reference Base can be maintained by a Merak Administrator or by a Spam Administrator, but it's highly recommended to leave the Reference Base as it is. Merak updates it automatically with the MIAS update and this database is based on processing (indexing and sorting) of millions of both spam and genuine messages.

Bayesian filters were considered to be unbeatable. Even if their accuracy on text messages is still very high, spammers have made many improvements to their message writing strategies and the importance of Bayesian filters in spam filtering (globally, not only the Merak implementation) is decreasing. We can show some examples when the Bayesian filter will totally fail:

```
--591.C50.D.EBA_6822__
Content-Type: text/html;
Content-Transfer-Encoding: quoted-printable

<p align=3D"center"><font face=3D"Geneva, Arial, Helvetica, sans-serif"><b=
><font color=3D"#FF0000" size=3D"5">Super Viagra and Generic Viagra</font><br>
</b>Both Available online with NO Dr. Visit!<br> <br> <b><font
size=3D"4">Cialis</font>, </b> also kn<!-- mcyuv -->own a= s Super Viagra or
the &quot;Weekend Drug&quot;<br> is like Viagra but Amazingly it works right
away and lasts 36 ho<!-- = jyvzsv cald oothx jb imrzwehn bjbhzcddgi
tefhvalcvtvmpgwf eqqnnkh -->urs!<br> <a href=3D"http://health-
products2003.com/index.php?KBID=3D1038">G<!-- = tp q u c dlpcnwcxyhk ylh
zqpvlyq p -->et it Here</a></font></p> <p align=3D"center"><font
face=3D"Geneva, Arial, Helvetica, sans-serif">Bo= <!-- dwvmahfjk g qw wiyzj
onv zz bzzhjd qswj pkisvrp fws zmxu zfc jo -->th products shipped discretely to
your door<br> <br> </font></p> <p>&nbsp;</p> <p>&nbsp;</p> <p><font size=3D"2"
face=3D"Geneva, Arial, Helvetica, sans-serif"><a href=3D= "http://www.health-
products2003.com/r">Please remove me</a></font></p> qkqb dwmni nci vxvintkdf
omdrfg uxvd ikh ofp iljtxjctoph nzykugwjftcaeb wacpfit
--591.C50.D.EBA_6822__--
```

Pretty unreadable, isn't it? The Bayesian filter engine will have the same confusion. The Bayesian filter will very likely not recognize it as spam. As we see, it will have problems with the HTML formatting and the nonsensical text that is commonly being added to the end of the message. It's hard to believe, but an even bigger problem for the Bayesian filter will be with meaningful text - some spam messages can have a portion of a conversation hidden at the end. That can overweight the probability computed from the first part of message. One last example shows a situation, where Bayesian filter don't have a chance at all:

```
=====
2501014361 373209557659 488613522379 554407932566
8663 7132 6035 8326 5772
2826 9831 8901 1299 7277
```

```

4935      8509      3071      8980      7667
4679292841 911649305203 594645563222 936350550374
0998      8872 5353      0864      6708
4325      4933      0590      2501      7564
1962      4783      3446      1378      3395
5663      6915      9507 276479617733 889759692477

      5511935348 260054058775 4253      4974
8671      11 3129      8870      8839
2176      27 0358      9690 4870
9544      3179      917855
      7577521514 930445051788 243131
      4291 8521      653539
64      6366 3245      3970 2454
40      6398 6129      5312      4012
      3099320598 792780270485 1601      5778

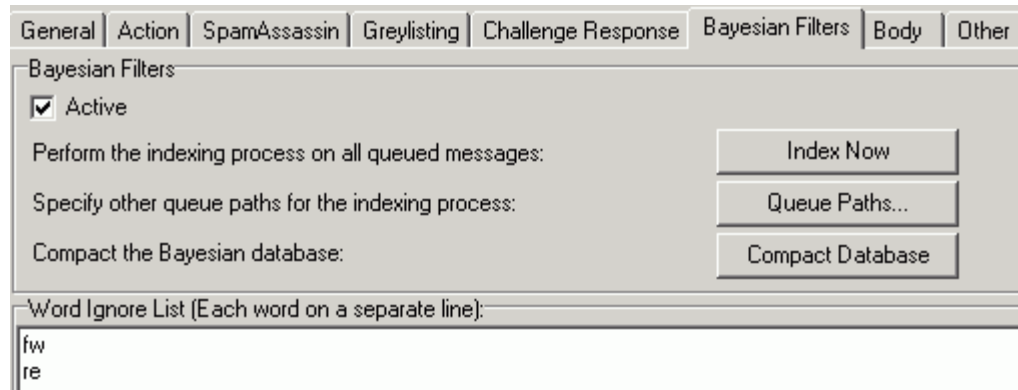
```

=====

Can you read that? I'm sure you can, but the Bayesian filter definitely can't(!) Still you should understand these examples as a warning that Bayesian filters are not all-powerful. But nothing is, far from that. As we mentioned before - all possible ways should be used against spam.

CHAPTER 10

Administration



Field	Description
Active	Enables the Bayesian filters. It is recommended that this option is enabled.
Index Now	<p>By pressing this button, the entire internal queue of Spam e-mail is indexed.</p> <p>This indexing creates a USER REFERENCE BASE that does NOT affect the main index which is automatically updated on its own.</p> <p>The indexing is handled in RAM MEMORY and is written to the User Reference Base when:</p> <ol style="list-style-type: none"> 1 the SMTP service is stopped, or; 2 At midnight when the scheduled indexing occurs

Queue Paths	<p>The "Queue Paths" button opens the file <code>/Merak/Spam/Spamqueue.dat</code> which is used for mapping of "Spam reference base folders". This is for additional Spam Reference Base Indexing.</p> <p>You can easily add an unlimited number of IMAP account directories or any other folders that you find suitable. There is no limit.</p> <p>The directories are handled the same as the core Spam directories. Once all messages are indexed, they will be deleted.</p> <p>Each folder mapping definition is located on a line by itself.</p> <p>Syntax:</p> <pre><directory>,<folder type>,<processing></pre> <p><folder type> = 1 : folder contains genuine mails <folder type> = 0 : folder contains spam mails <processing> = 1 : deindexing <processing> = 0 or nothing : indexing</p> <p>Examples:</p> <p>Folder for indexing Spam mails <code>MailMerakdemo.com\IMAP\Spam;1</code></p> <p>Folder for indexing Genuine mails <code>MailMerakdemo.com\IMAP\Genuine;0</code></p> <p>Folder for correction of Genuine e-mail that has been indexed as Spam <code>MailMerakdemo.com\IMAP\Spam-Genuine;0;1</code></p> <p>Folder for correction of Spam e-mail that has been indexed as Genuine <code>MailMerakdemo.com\IMAP\Genuine-Spam;1;1</code></p>
Compact Database	<p>By pressing this button, you will remove words that occur at a low frequency. These words are mostly random words that you usually see included in Spam e-mail.</p> <p>By compacting your database, the accuracy of the Bayesian filter will increase because these low frequency words have been removed.</p> <p>Only the "User Reference Bbase" is compacted by this button.</p>
Word ignore list	<p>Contains the words that will be ignored during the Spam Reference Base update (indexing process). We highly recommend that you propagate this with words that are often used in your own internal communications such as company name, products, services etc.</p> <p>You can and should include words that are native to your particular type of business.</p> <p>However, if you intend to use the Automatic Bayesian Filtering update, you can ignore this altogether.</p>

CHAPTER 11

Configuration Files

Bayesian filters use the MIAS configuration file, but it also have a special file (spam\spamqueue.dat) for customization of its indexing process. The spamqueue.dat file syntax is described in the example file examples\spamqueue.dat.html. The indexing process is pretty well configurable by either using this file or properties in the MIAS configuration file (spam\spam.dat). You can limit the count of the message to be queued for indexing per day by setting the `MaxIndexMessagesPerDay` property, or you can define spam and genuine messages by specifying the limit percentage score (`IndexGenuinePercentage` and `IndexSpamPercentage`). There are many more options that could be configured using properties in both files, but it's highly recommended that you do so as your own risk. If you have the Bayesian reference database updated by either an automatic update process or by manual update, you should not need to perform an indexing process. Performing indexing on your own can actually harm the sensitivity of your Bayesian engine.

The only really important property in the spam.dat file for Bayesian filters is the `SpamBayesian` field that has to be set to 1 if you want to enable it.

CHAPTER 12

Body & HTML filters

In This Chapter

How It Works	60
Administration	62
Configuration Files	66

CHAPTER 13

How It Works

Body & HTML filters

Spam messages are very often HTML formatted. These messages are sometimes hard to recognize by Bayesian filters, because they use tables to format text. For example, it may contain the word *viagra*, but with invisible formatting `v|i|a|g|r|a`. The SA engine won't find that text, but user will read it without a problem. MIAS contains some tests that can help catching such messages. However, many companies use only HTML formatted messages for legitimate emails and can find problems with high count of false positives. In such cases you shouldn't use these tests or you have to bypass trusted domains.

So, which tests are available?

- § Score HTML messages with present text/html but missing text/plain part: Messages from regular mail clients that are written with HTML, should always have the same content written in plain text form as well. In most HTML Spam e-mail, the text part is missing.
- § Score HTML messages with external URL: The external URL in HTML is not a normal part of most messages but is typical for computer generated ones.
- § Score HTML messages containing no text: Some Spam messages simply have no subject or a URL as the body. This is also not typical for the legitimate e-mail. You can leave this option on even if you frequently send such e-mail - see the white listing strategies *at Black & White Listing* (see "Black and White Lists" on page 73) techniques
- § Score HTML messages containing script code: Messages with script code in the HTML are also not usually created by a regular e-mail client and scripts contained in e-mails can often be malicious.
- § Score HTML messages with different content or text/plain and text/html parts: For this feature, the text and html part of the regular messages must match exactly. If not, it is considered to be Spam. There are some situations and clients when this rule is not always fulfilled, however, together with the *Black & White Listing* (see "Black and White Lists" on page 73) techniques you can use this option while still maintaining a high level of accuracy. Actually this is exactly the right test that would help us with the example above.
- § Score messages containing no subject and no body: Regular messages usually contain subject and some entries in the message body. If you are used to send/receive such messages leave the check-box unchecked.

Charset filters

You can discover that your company is getting a lot of messages in foreign languages using odd or just less common characters. MIAS offers some tests that can help you catch these messages:

- § Forbidden Charsets: You can choose which charsets charsets are forbidden on your mail server and all messages using these charsets will be punished with some score (that is also configurable).

- § Score messages with missing charsets and characters higher than 0x7F: If the information about the charset is completely missing in a message header - it is probably a message generated by some automated send mail system. Regular e-mails not written in English do use characters with a value bigger than 127 often (reflecting the local character set), but legitimate email client should add the charset information into the message header.

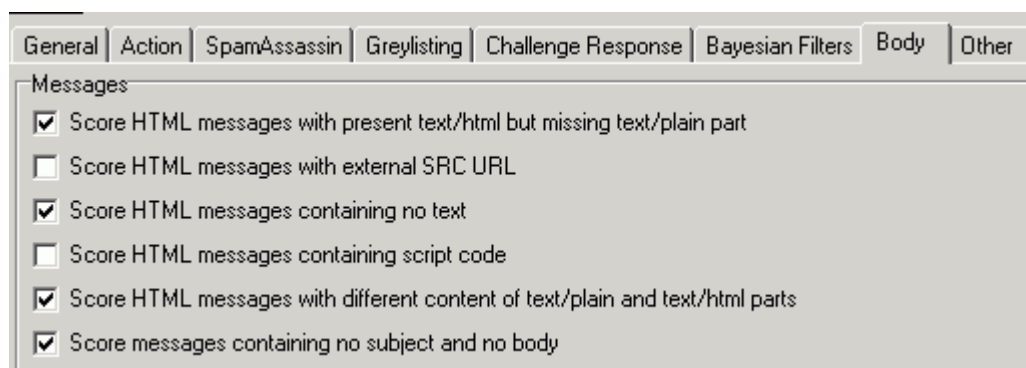
CHAPTER 14

Administration

These filters were developed to "catch" most of the automatically generated emails, which are usually provided by some active-x components (automated Spam mailers). These emails are mostly different from the ones created by regular clients.

These filters are very effective in blocking most of the "html based spam", which are very popular now.

If you are creating HTML e-mail in your organization because it is the nature of your business, simply set the bypass for your senders so that MIAS is not applied to outgoing messages.



Fields	Description
Score HTML messages with present text/html but missing text/plain part	<p>Messages from regular mail clients that are written with HTML, should always have the same content written in plain text form as well. In most HTML Spam e-mail, the text part is missing.</p> <p>This option will score these messages and the score is added to the total score of the message.</p>
Score HTML messages with external SCR URL	<p>The external SCR URL in HTML is not a normal part of most messages but is typical for computer generated ones.</p> <p>This option will score these messages and the score is added to the total score of the message.</p>
Score HTML messages containing no text	<p>Some Spam messages simply have no subject or a URL as the body. This is also not typical for legitimate e-mail.</p> <p>You can leave this option on even if you frequently send such e-mail - see the white listing strategies at Black & White Listing Techniques</p> <p>This option will score these HTML messages and the score is added to the total score of the message.</p>
Score HTML messages containing scrip code	<p>Messages with script code in the HTML are also not usually created by a regular e-mail client and scripts contained in e-mails can often be malicious.</p> <p>It is good idea keep this option on and to score up the total message score.</p>

Score HTML messages with different content or text/plain and text/html parts	<p>For this feature, the text and html part of the regular messages must match exactly. If not, it is considered to be Spam.</p> <p>There are some situations and clients when this rule is not always fulfilled, however, together with the Black & White Listing techniques you can use this option while still maintaining a high level of accuracy. The checked option will score up.</p>
Score messages containing no subject and no body	<p>Regular messages usually contain subject line and message body content. If you are used to send/receive such messages that have either of these parts empty, leave the check-box unchecked.</p> <p>If you consider messages to be spam, leave this option checked and let Merak Instant Anti-Spam to score up the total message score.</p>

Characters

Forbidden charsets:

Score messages with forbidden charsets

Score messages with missing charsets and characters higher than 0x7F

Field	Description
Forbidden Charsets Score messages with forbidden charsets	<p>Sometimes the best protection against unwanted e-mails is rejecting e-mails written in certain languages. This recognition is determined by checking for this feature used in the message header.</p> <p>You can set charsets for all languages you do not want to accept. See the table with charset names below.</p> <p>The default setting in Merak, is set to score up anything in Simplified or Traditional Chinese. Score is added to total message score.</p>
Score messages with missing charsets and characters higher than 0x7F	<p>If the information about charset is completely missing in a message header - it is probably a message generated by some automated send mail system.</p> <p>Regular e-mails are also not using characters with a value bigger than 7F (hexa). Even mails with languages using special ASCII characters (French, Spain, etc.) are encoded values lower than 7F.</p> <p>Mails with these characters are scored and the score is added to the total score.</p>

Important note: If you are sending through Merak Mail Server, forms from your web site, they can sometimes contain some characters with are ASCII values greater than 127 (usually foreign names). To prevent the marking of such e-mails as Spam, be sure to always define the proper value for the "character set" in the object that is used for processing the form and white list your computer with the send mail component.

Charset	Charset Code
Arabic (ISO)	iso88596
Arabic (Windows)	win1256
Baltic (ISO)	iso88594
Baltic (Windows)	win1257
Cyrillic Alphabet (ISO)	iso88595
Cyrillic Alphabet (Windows)	win1251
European Central (ISO)	iso88592
European Central (Windows)	win1250
European Western (ISO)	iso88591
European Western (Windows)	win1252
Greek (ISO)	iso88597
Greek (Windows)	win1253
Hebrew (ISO)	iso88598
Hebrew (Windows)	win1255
Chinese Simplified (GB2312)	gb2312
Chinese Traditional (Big5)	big5
Japanese (EUC)	euc-jp
Japanese (ISO)	iso2022jp
Japanese (SHIFT_JIS)	shift_jis
Korean (EUC)	euc-kr
Korean (ISO)	iso8088kr
Korean (ks_c_5601-1987)	ksc56011987
Turkish (ISO)	iso88599
Turkish (Windows)	win1254
Unicode (UTF8)	utf8

CHAPTER 15

Configuration Files

Use the standard MIAS configuration file (`spam\spam.dat`) for a complete configuration. Relevant properties are these:

Variable	Type	Description
<code>CharsetSpam</code>	Boolean	When set to 1, some points are added to spam score of every message that uses forbidden charset (see below).
<code>CharsetSpamScore</code>	Float	Defines a score that will be added to messages using forbidden charset.
<code>SkipIndexingForCharsets</code>	String	This defines forbidden charsets. Use standard charset code (e.g. win1252). When specifying more charsets, use semicolon (;) as delimiter.
<code>MissingCharset</code>	Boolean	Setting this to 1 enables scoring messages that uses characters of ASCII code val 128 or higher while don't have explicit charset definition in header.
<code>MissingCharsetScore</code>	Float	Defines score added to a messages missing charset definition but using characters with ASCII code 128 or higher.
<code>SpamHTMLPartsOnly</code>	Boolean	Enables scoring of HTML formatted messages that are missing text/plain part.
<code>SpamHTMLPartsOnlyScore</code>	Float	Score that is added to HTML formatted messages without text/plain part.
<code>SpamHTMLExternalURL</code>	Boolean	Enables scoring of messages containing external URLs.
<code>SpamHTMLExternalURLScore</code>	Float	Specifies the score added to every message with external link.
<code>SpamHTMLNoText</code>	Boolean	Setting this to 1 enables scoring of HTML messages that contains no text
<code>SpamHTMLNoTextScore</code>	Float	This defines the score applied on HTML messages with no text.
<code>SpamHTMLScript</code>	Boolean	Set to 1 to enable scoring of messages that contains script code in HTML part.
<code>SpamHTMLScriptScore</code>	Float	Score used for messages that contains script code in their HTML part.
<code>SpamHTMLEqual</code>	Boolean	By setting this to 1 you enable comparing of text/HTML and text/plain parts of th message.
<code>SpamHTMLEqualScore</code>	Float	This score is added to messages with different content of text/HTML and text/pla parts.
<code>SpamBlankMail</code>	Boolean	When set to 1, messages with no text in subject and no body are scored up.
<code>SpamBlankMailScore</code>	Float	Specifies the score that is added to messages with no text in subject and no body.

Content Filters

CHAPTER 16

How It Works

Content filtering is the most powerful filtering method. It contains a wide variety of tests not based only on message headers and content. You can even set up very complex filters by compounding several simple rules and joining them by logical operators AND and OR. However, you have to be cautious, since complex filters can take too much time for processing big messages. It's highly recommended that you set up complex filters with additional conditions for maximal message size. Also if you are creating filters for inbound messages, you can append a remote-to-local test to your filter so it doesn't process all messages coming through your server.

For the complete list of conditions available for content filters, please consult Merak manual or context help. The actions of some content filters can vary. This is also the strong side of content filtering. You can simply accept the message (remove preceding rejection), mark it as spam/genuine or delete it, but you can also perform other actions like sending a new message, adding headers to the message, write data to some file (perform custom logging) or even tarpit the sender. Moreover just as you can set more conditions to one filter, you can also set more actions to one filter. This gives you the possibility of deleting an incoming message, sending information to your administrator and adding an entry to some log file all in one easy-to-create filter!

As you already know, content filtering is applied after MIAS processing. This (and a wide variety of CF conditions) gives you the possibility to use MIAS results as parameters for content filtering. So you are able to not only perform filtering by CFs, but you can also manage messages already processed by your antispam engine. Here are the conditions you can use:

Where spam engine detects spam

This condition has no parameters and it only checks for the spam flag. Even if MIAS itself allows you to perform the most simple actions (place message to spam folder, forward or delete), this condition can still be very useful. For example, you would like to accept spam messages coming to one particular domain, but you want to log them separately to some file and so you can't remove that domain from MIAS processing.

Where SpamAssassin score is value


There are only three score limits that can be configured in MIAS allowing you to mark a message as spam, quarantine or delete it. However, you may want to perform some complex sorting depending on the SpamAssassin score. In this case, there is a CF condition which accepts a parameter of the score limit and all messages with a higher or lower (this depends on the filter setting) SA score are filtered out by this filter.

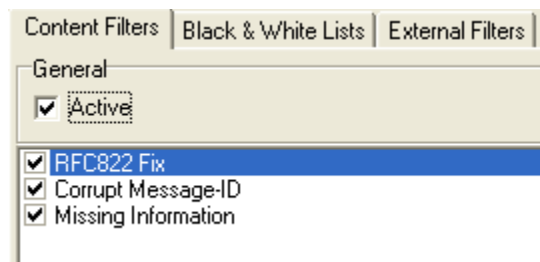
Where Bayesian probability is percentage

Even if the spam probability computed by Bayesian filters is converted into score value that is being added to the MIAS total sum as we already know, the percentage values stored with the message could be used by this condition. You can choose the percentage limit for the filter and whether messages with lower or higher values should be filtered.

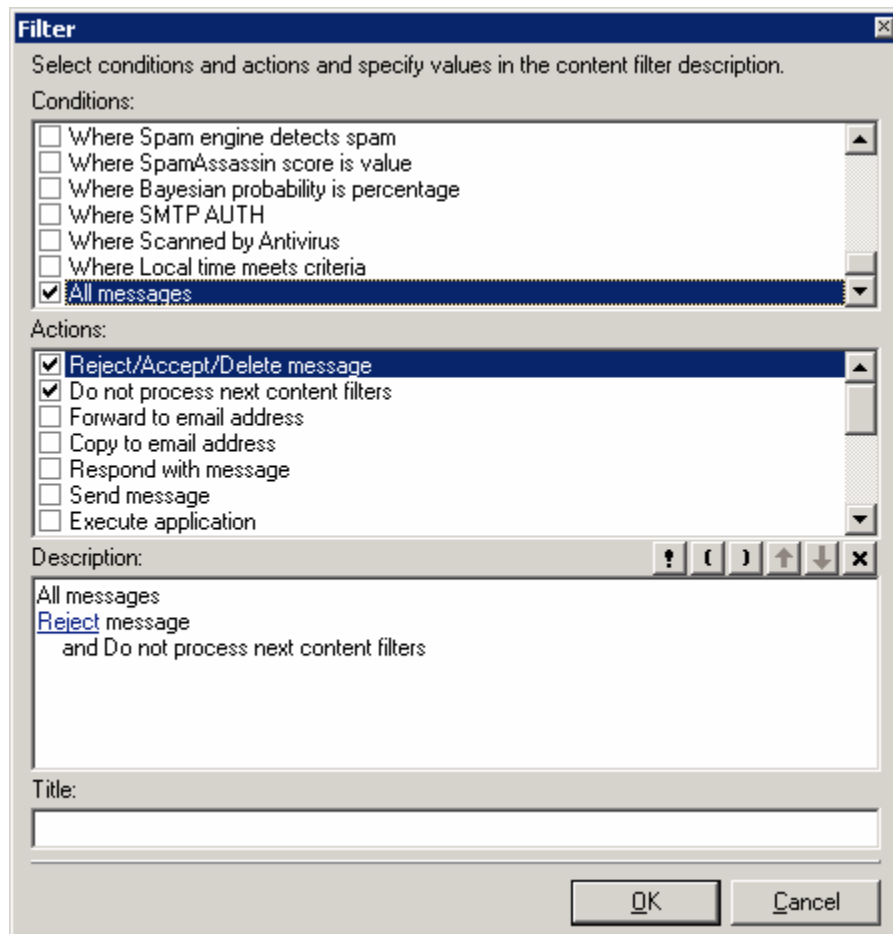
CHAPTER 17

Administration

The Content Filters section of Windows Merak Administration Console shows a sorted list of all content filters set up on your mail server. There is a checkbox beside every filter which you can use to filter (de)activation. The entire content filtering process can be bypassed by the definitions in the bypass file (see  button) and by unchecking the Active checkbox you can turn off content filtering all together. See the image below.



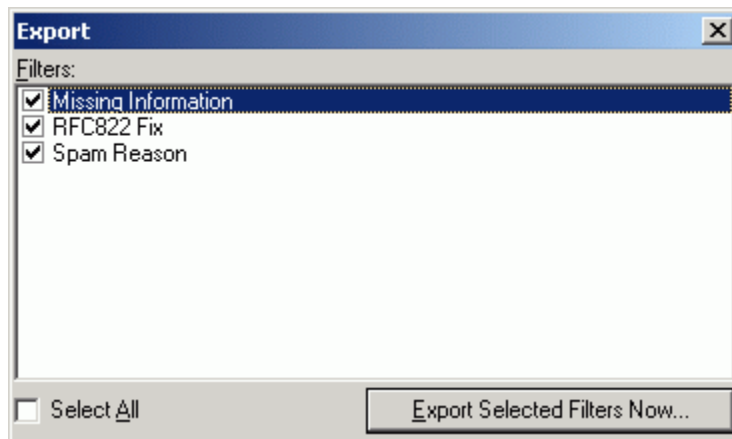
By doubleclicking on any existing (active or inactive) filter or by clicking on the "Add" button, a content filter properties dialog pops up. You can compound the filter by adding conditions using logical operators or by adding/removing actions. The image below shows this dialog.



Notice that there is a "Title" field in the content filter properties dialog. This is useful in list view of all content filters so you know which content filter you working on. The CF title is also used in the SMTP response returned to the sender when the message is rejected (and also appears in SMTP log). It is recommended (but not necessary) to write a brief description into this field.

Content Filters Export / Import

Content filters can be imported/exported to/from an XML file using the Import and Export buttons. Content filters can have a unique ID and "read only" status. In these cases, you can view the filter item and its attributes will be displayed above the filter title on the right side. Unique IDs are used when importing filters, so the IDs are validated against each other. Read- Only filters cannot be edited in any way.



We are not about to explain all conditions and actions that content filters offer, however an incomplete list would be less useful. So please, read the official Merak Mail Server administration guide or check out the context help for Merak Administration Console for complete list of all content filter features.

CHAPTER 18

Configuration Files

All content filters are stored together in one XML formatted file `config\content.xml`. This file is easy to read, however all conditions and actions are specified by their ID number and thus it is impossible to edit the file without the Merak Administration Console.

Use `config\cfbypass.dat` file as the bypass file for content filters.

Black and White Lists

CHAPTER 19

How It Works

Black & White filters are a very complex way of filtering messages. It offers both a long list of conditions and a variety of actions that can be performed. As was introduced before, these filters occur in two places in message processing. First, after an SMTP session starts when the message sender and recipient are announced. This group of filters provides tests against SMTP sender and recipient and sending server IP. The second group of filters (rules) is applied when the whole message is received. It can perform tests based on the message content.

Filtering is performed on three levels:

- § Global - for the entire server
- § Domain - for each individual domain
- § User - for each individual user

This means, that you can set a rule that is relevant for all messages coming thru your server (global B&W rule), but you can also create a rule that is applied only on messages coming to one particular user. The power of three filtering levels is not just in the filters relevancy that could be also reached by setting up more complex filters, but it's also hidden in the fact that user filters can be set by user and domain filters can be set by domain administrators.

Now we can briefly discuss all conditions available for any B&W rule:

- § All - This condition will always be true.
- § Any header - All headers of the message will be searched for the specified string.
- § Specified Header - Specified header is searched for a string.
- § Body - Checks the whole body for some content.
- § Attachment - A condition for attachment names.
- § Size Greater - Checks if the message size is greater or equal than the given number of Kbytes.
- § Size Lower - Checks if the message size is lower or equal than the given number of Kbytes.
- § Spam - Checks if the message was marked as spam.
- § IP Address - The IP address item lets you check the IP address of the originating server the message was sent from.
- § SMTP Sender - Lets you check the envelope sender of the message.
- § SMTP Recipient - Lets you check the envelope recipient of the message.
- § SMTP AUTH - A condition which is true only if the session from the originating server used the SMTP authentication.
- § rDNS - Makes sure the originating mail server has the PTR record.

As you see, there are a couple of items you can check in the message. Of course, you can search for an exact string, a regular expression or even a list of strings specified by some text file (usable for address matching).

There are also several actions Merak can perform when the condition is fulfilled. You can

- § Reject the message (=blacklisting)
- § Accept the message (=whitelisting - clears the spam flag and bypasses all following filters)
- § Delete the message completely
- § Mark the message as spam
- § Forward the message
- § Send a new message
- § Forward the message to IM account

B&W filters are much like content filters, but are not so strong. They can be used by any user and are simple to set. This fact makes them very useful. The most important feature is that some B&W filters are applied before DATA command and unnecessary data traffic.

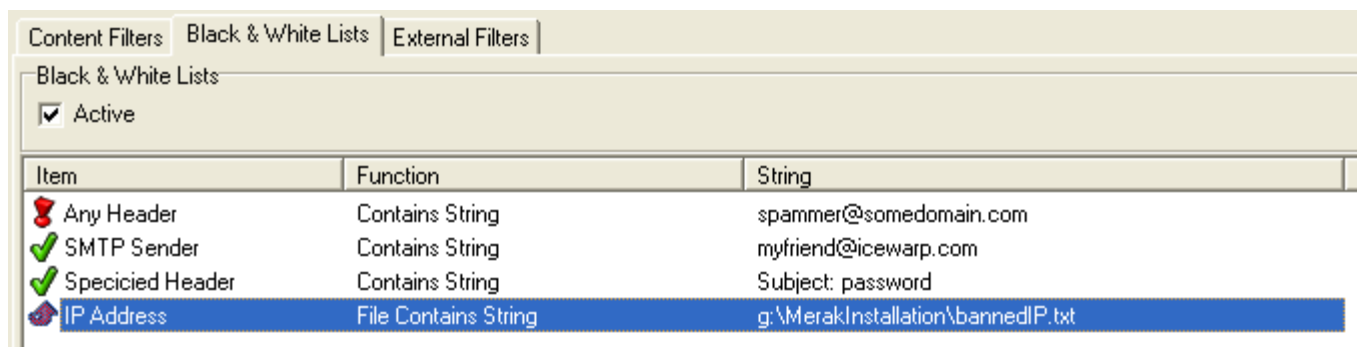
Remember, that as much as the blacklisting is important for rejecting spam and messages from spammers, one should use whitelisting to prevent your friend's (partner's, customer's) messages from being filtered out as spam! Remember that advertisement messages from your business partners will be caught by MIAS if you don't whitelist the sender.





CHAPTER 20

Administration

The settings are the same for all three levels. You can find Global B&W filters in Mail Service -> Filtering node and Domain and User B&W filters in B&W List tab of appropriate domain/account properties.

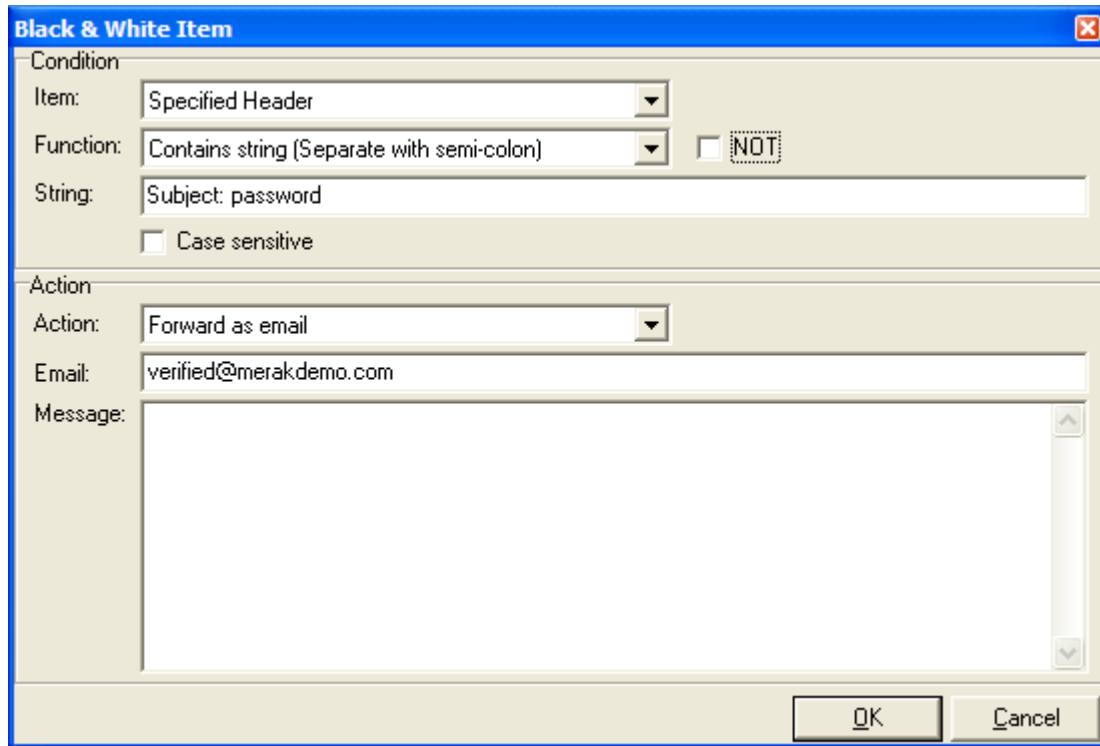
Global Black & White List



Item	Function	String
 Any Header	Contains String	spammer@somedomain.com
 SMTP Sender	Contains String	myfriend@icewarp.com
 Specified Header	Contains String	Subject: password
 IP Address	File Contains String	g:\Merak\Installation\bannedIP.txt

To define a new entry, click on the Add button and follow the dialog. All entries defined there will affect the entire server (all users).

The full description of this window is located in the interactive help by pressing "F1" while in the console.



Black & White Item

Condition

Item: Specified Header

Function: Contains string (Separate with semi-colon) NOT

String: Subject: password

Case sensitive

Action

Action: Forward as email

Email: verified@merakdemo.com

Message:

OK Cancel

User and Domain B&W filters are the same, only their effect is different because it affects only the appropriate domain/account. Remember, that any server rule can be overruled by any rule on a domain level. Similarly, user rules will overrule domain rules..

Entries in the user's "B&W List" can be created by:

- § The Merak Administrative interface (see above)
- § Merak WebMail Interface
- § Sending/Redirecting mail to a special List Server account
- § Instant Messaging Client

CHAPTER 21

Configuration Files

The configuration is performed on three levels and each level uses a different file. Server B&W list - the global one - is stored as `config\filter.dat` file. The bypass file you can use for bypassing the global B&W filters is called `config\gabypass.dat`.

Domain and user filtering is performed per domain or account respectively, and thus a separate file is used for every domain/account. For domain filtering use the file `config\[domain]\filter.dat` (change the [domain] with appropriate domain name) and `config\dabypass.dat` for bypassing. Notice, that there is no special bypass file for each domain, rather an universal file is used for all domains. The same situation exists for user accounts. User B&W lists are stored in `mail\[domain]\[user]\filter.dat` and uses `config\gabypass.dat` as a bypass file.

Please consult the example file `filter.dat.html` in the examples folder for precise syntax of any B&W list file.

Other

Merak itself, outside of the Instant AntiSpam engine, contains some features that can help you with the blocking of spam. These are tightly bound to the SMTP relaying configuration. Check out the following topics for those most important and useful.

CHAPTER 22

Tarpitting

How it works

Tarpitting monitors all unsuccessful attempts to deliver messages to unknown users and if the number of such attempts exceeds the count limit, the IP address of the sender will be tarpitted for a period of time and no access from that IP address will be allowed.

This option serves as a protection from spammers trying to spam your mail server accounts based on email address dictionary attacks. This function is even more useful when you configure it to consider relay rejected sessions to be unsuccessful attempts too or if you enable tarpitting for server that exceed some connection limit per one minute.

The real significance of this feature is that when it is already activated against some server, it rejects every connection attempt immediately without any communication.

It has a small flaw that must be considered. A long list of tarpitted addresses can damage message processing speed significantly.

Administration

Merak offers a feature called tarpitting. Tarpitting monitors all unsuccessful attempts to deliver messages to unknown users and if the number of such attempts exceeds the count limit, the IP address of the sender will be tarpitted for a period of time and no access from that IP address will be allowed.

This option serves as a protection from spammers trying to spam your mail server accounts based on email address dictionary attacks.

When an IP address has been tarpitted, you might require that IP to communicate with your server again in the future. For this purpose you can use the bypass feature.

The screenshot shows a configuration window with four tabs: "Anti Relaying", "Protection", "Tarpitting", and "Protocols". The "Tarpitting" tab is selected. The window contains the following settings:

- Active
- A number of attempts after which the IP address will be tarpitted: 0
- An amount of time for IP addresses to be tarpitted (Min): 25
- Cross session processing

Field	Description
Active	Enables this feature.
A number of attempts after which the IP address will be tarpitted.	Here you can define the number of attempts you needed to be tarpitted.
An amount of time for IP addresses to be tarpitted.	How long the IP address stay tarpitted (so the connections from that IP will be automatically rejected)
Cross session processing	<p>When enabled, the number of attempts is recorded from multiple sessions together. This requires the server to reserve some memory for such behavior. If not checked only the current session attempts are considered.</p> <p>It counts attempts from different sessions for the same time interval that is specified in "An amount of time for IP addresses to be tarpitted" option.</p>

Action

Block tarpitted IP addresses

Close tarpitted connections

Field	Description
Block tarpitted IP addresses	<p>Checking this option will store the tarpitters IP and block it. This feature is suitable for spammer's IP addresses that change a lot.</p> <p>However, the unchecked state lets you use the tarpitting feature but the Tarpit IP DB is not used, you can force such sessions to be closed only.</p>
Close tarpitted connection	If an IP address is to be tarpitted the connection will be also closed.

Other

Enable tarpitting for relay rejected sessions

Tarpit IP addresses that establish a number of connections in 1 minute:

Field	Description
Enable tarpitting for relay rejected sessions	This option considers relay rejected sessions to be unsuccessful attempts too.
Tarpit IP addresses that establish a number of connections in 1 minute	This option will tarpit all IP addresses that exceed the limit of this setting. That means the IP addresses establish a high number of connection in 1 minute.

Tarpitted IPs	
IP	Time Stamp
81.0.232.34	2005/01/28 14:17:59

Refresh list so to see all tarpitted IPs here (and when it was tarpitted) and if there is any you do not want to be tarpitted anymore, just use the Remove button in the bottom. Use the Remove All button to remove all tarpitted addresses or Remove All Expired to remove tarpitted addresses which expired.



For adding an IP to the tarpit list, simply click the Add button and fill in the desired IP or IP range. This newly added IP will obey the general rules for tarpitting as defined in the settings above on this screen.

Configuration files

The file which contains tarpitted servers (config\tarpit.dat) is not of a text format and so it is not so easily readable/editable, thus configuration of tarpitting is not possible by editing some file manually. You have to use either the Merak Administration Console for Windows or web based administration interface.

The bypass file can be edited however as it is of a common format. You should add trusted servers into it: config\tarpitbypass.dat

CHAPTER 23

DNSBL

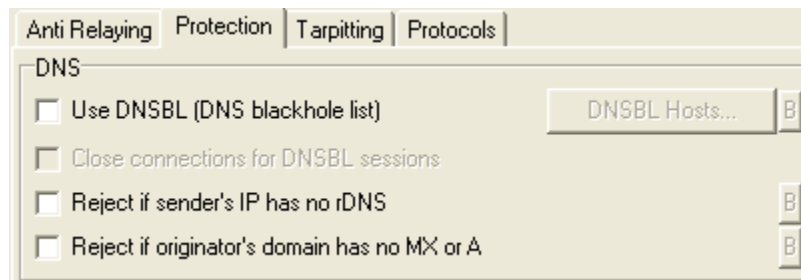
How it works

A DNS black list is a service which provides a list of known spammers' IP addresses. The DNS server is used as a database or directory mechanism to store their IP addresses along with some other information. It can also be used as a DNSWL where the DNS server stores the IP addresses of genuine senders. In such cases you can use the DNS server for white listing and black listing purposes.

A free public service is very easy to use, you just have to choose some existing DNSBL server and specify it in Merak configuration. When the server SMTP session starts, Merak performs a query for each DNSBL server specified. The exact functionality is similar to B&W listing, thus when some DNS BL server entry matches the sender, the message is rejected. When some DNS WL server entry matches the sender, the message is accepted.

You should use the smallest number of DNSBL servers possible. Too many records might slow down a message delivery due to too many DNS requests.

Administration



Field	Description
Use DNSBL	<p>A DNSBL is a service which provides a list of known spammers' IP addresses. The DNS server is used as a database or directory mechanism to store their IP addresses along with some other information.</p> <p>It can also be used as a DNSWL where the DNS server stores the IP addresses of genuine senders. In such cases you can use the DNS server for white listing and black listing purposes.</p> <p>You have to specify your own DNSBL servers in the editor dialog. Follow the example file information. You should use as least as possible of multiple DNSBL servers. Too many records might slow down a message delivery due to too many DNS requests.</p> <p>BL servers</p> <p>All servers without a special prefix are black list servers and all session IP addresses on the DNSBL server will be rejected. This is the most used way for DNSBL servers.</p> <p>WL servers</p> <p>Server specifications with a special prefix will act as white list servers. All session IP address on the DNSWL server will be accepted.</p>
Close connections for DNSBL sessions	This option lets you automatically close all server sessions for IP addresses found on the DNSBL server.

Configuration files

DNS blacklisting and whitelisting servers can be actually set up only by editing the config\rbl.dat file, so you have to know the exact syntax: One DNS BL server per line. If you want to use the server for whitelisting, just start the line with string "1:". An example follows:

```

dnsbl.njabl.org
list.dsbl.org
1:whitelist.server.org ; whitelisting DNS server

```

As you can see, you can even use line comments separated by a semicolon from the server address. This comment is very useful because it will appear in the SMTP session log when some entry returned by that particular DNS server was applied for that SMTP session. This way you have at least light control over the process.

Trusted servers can be added into bypass file `config\rblbypass.dat`.

CHAPTER 24

Miscellaneous

Here are some useful features that can help you in filtering unwanted messages and/or connections. These don't have to be explained so thoroughly. In the Merak Administration Console you can find these features in the Mail Service configuration under the Security section.

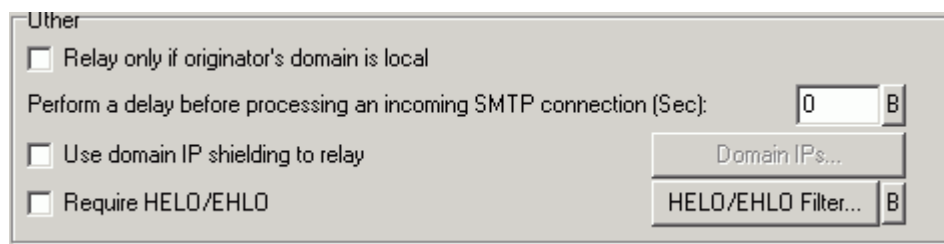
HELO/EHLO

Enabling the option "Require HELO/EHLO" makes the mail server require the HELO or EHLO command to be present in the SMTP session. If not, such sessions will be rejected.

Moreover, you can specify in a special file `config\heloehlo.dat` which hostnames are allowed and which are not. The file is a filter for the name of the server that is specified during the HELO/EHLO command in an SMTP session so you can easily block some servers without knowing their actual IPs. The syntax of this file is very simple: every filter has to be on separate line, starting with "0:" or "1:" - 0 for rejecting the server, 1 for accepting. You can use masks. See the example below:

```
1:mail.icewarp.com
0:*spamsite.com
0:mail.spammer.com
1:mail.yahoo.com
```

You can use a bypass file `config\heloehlobypass.dat`, which respects the common syntax for bypass files.



Delayed SMTP session processing

This very simple feature stands as a light version of greylisting. A value greater than zero will perform a delay before processing an incoming SMTP connection. The value is in seconds.

Spammer's servers usually disconnect after a short period of time. Legitimate servers wait for a longer time period. This option is also tied with "Trusted IPs" list, so these IPs are bypassed. Still if you want to bypass more servers that are not set as 'trusted', you can do that by editing the bypass file `config\smtpwaitbypass.dat`.

Recommended value for this feature is 30 (seconds).

CHAPTER 25

Troubleshooting

In This Chapter

Engine Logging	88
SMTP Test Tool.....	92
FAQs	94

CHAPTER 26

Engine Logging

Once you get MIAS working you will sooner or later ask a question whether it is doing what is supposed to do. Sometimes a question will arise: "Why is (not) this message in my spam folder?" Then you will find the MIAS logging function to be useful. It is worth reading the next few lines for a description of this feature.

First, we will show one sample of a log entry and then we will discuss every single character:

```
SYSTEM [00001184] Tue, 6 Dec 2005 12:05:27 +0100 JXL03127
'<roman@icewarp.com>' '<flavio@lucanet.com.br>' score 1,86 reason [Body=01]
action NONE
```

Every entry is just a single line. MIAS logging tries to keep the format used for other Merak logs. That's why every log entry will start with "SYSTEM" text. The next field in square brackets shows the thread ID. As well as the last one, this field is for compatibility or the possibility of debugging; you will hardly find a use for it otherwise. These uninteresting fields are followed by date and time when the message was processed by the MIAS engine. Its format is pretty obvious from the sample. The fourth field displays engine response. It should return "+0100" always for confirmation that everything passed alright. The message ID follows. This will be a very important field once you open the SMTP log file and want to find an appropriate entry in there.

Notice that some fields, even if they seem to be useless for you, are necessary for some log analyzers that are already available and will be able to read the MIAS log file without any changes.

The next two fields tell you who was the sender and who was the recipient in this order. In the example above "roman@icewarp.com" sent a message to "flavio@lucanet.com.br". Next item shows you the SCORE that this message reached in the MIAS scoring system (mainly the Spam Assassin score). In our example the message was scored with 1,86 points which wasn't enough to mark it a spam as we will see in a while.

When the message is scored by some filter, the filter is displayed in the list of reasons. The list is bracketed, because it can contain several items. However, as you can see there was just one test failure in our example. Only one Body test failed and thus increased our message score. The last item specifies the action that was performed on the basis of the final score (1,86) and MIAS configuration. The engine in our example was set to flag messages with score 3,00 or higher (this can't be read from the log file), which wasn't met and thus the message was considered to be genuine. The "NONE" text reflects this situation.

Actions

Now, have a look at the following examples that show all possible actions that can occur. Match these actions with appropriate message scores that were reached. (Spam limits were set to 3, 5 and 9 for "mark as spam", "quarantine" and "delete" in this order.)

```
SYSTEM [00001184] Tue, 6 Dec 2005 11:51:35 +0100 JWF21835
```

'<sender@domain.com>' '<recipient@domain.com>' score -4,39 reason [] action NONE

SYSTEM [00000DCC] Sun, 4 Dec 2005 13:43:39 +0100 HYV17139
'<bypassed.sender@domain.com>' '<recipient@domain.com>' score 0,00 reason [Bypass=20] action BYPASS

SYSTEM [00000DCC] Sun, 4 Dec 2005 15:55:33 +0100 HAH81233
'<sender@domain.com>' '<recipient@domain.com>' score 6,27 reason [SpamAssassin=6,27] action SPAM

SYSTEM [00001184] Tue, 6 Dec 2005 17:10:07 +0100 JCQ46806
'<sender@domain.com>' '<recipient@domain.com>' score 9,57 reason [SpamAssassin=9,57] action DELETE

SYSTEM [00001184] Tue, 6 Dec 2005 17:10:59 +0100 JCQ51559
'<sender@domain.com>' '<recipient@domain.com>' score 5,26 reason [SpamAssassin=5,26] action QUARANTINE

As you can see, there are 5 possible actions that can be performed:

- § NONE action occurs when even the lowest score limit for marking message as spam wasn't reached
- § SPAM action occurs when the score limit for marking message as spam was reached, but not any higher
- § QUARANTINE action occurs when quarantine limit was reached
- § DELETE action occurs when the score limit for message deletion was reached
- § BYPASS action occurs when the message was bypassed for MIAS engine (that means the sender or recipient matched with some bypass file entry)

Remember, that the displayed action is somewhat like the recommendation made by MIAS. The final action is performed by Merak and thus other filters can contribute to the decision. Thus even if the action is SPAM, the message could be whitelisted and therefore delivered to recipient's inbox. And vice versa - the logged action could be NONE, but the message could be deleted by content filter.

Reasons

Now we should enumerate all reasons that can appear in the reasons list:

SpamAssassin=score

Spam Assassin score is always computed and can be positive or negative. However, it is listed here only if it reaches some lower limit that could be enough for marking a message as spam. The score with which Spam Assassin contributed to the total sum is displayed so that you know how important its result was and what was the contribution of the rest of filters.

Bypass=flags

When some bypassing rule is applied, the action is set to "BYPASS" and in the reasons field the "Bypass" item appears. There is more than one bypass file and so a hexadecimal number is used that (when converted to binary form) specifies all bypass rules that were used (that contained a matching item). For example, the log entry looks like this one: Bypass=24. We have to convert this hexadecimal number to binary (24=00100100) and then check which bits are set to 1. Every bit is associated with some bypass file. The association is as follows:

Bit code	Meaning
00000001	Occurs when you run out of MIAS license and so the filter processing is not performed at all.
00000010	Message was whitelisted.
00000100	Bypass trusted IP addresses and authenticated session.
00001000	This bypass is applied on outgoing messages only, when you have AntiSpam configured to exclude outgoing messages from processing.
00010000	Messages size limit was applied, preventing big messages from being processed by MIAS. This limit can be set in spam.dat file (96 kB by default).
00100000	MIAS bypass file: spam/spambypass.dat.

- 01000000 A non-user account is the recipient of a message and processing of these accounts is not enabled.
 10000000 Processing mode or account service access is set so that MIAS is not applied for this account.

Body=flags

The same logic as is used for when a Bypass item is used for Body (HTML filters). So the hexadecimal value has to be converted to binary form and then you can directly read all used filters. The bit association to filters reflects the order in which these filters are displayed in Merak Administration Console associating the lowest bit (the rightmost) to the first filter:

Bit code	Meaning
00000001	Score HTML messages with present text/html but missing text/plain part
00000010	Score HTML messages with external SCR URL
00000100	Score HTML messages containing no text
00001000	Score HTML messages containing scrip code
00010000	Score HTML messages with different content or text/plain and text/html parts
00100000	Score messages containing no subject and no body

Charset=flags

This uses the same logic as Body flags. Association reflects the Administration Console. There are only two charset filters that can be applied:

Bit code	Meaning
00000001	Score messages with forbidden charsets
00000010	Score messages with missing charsets and characters higher than 0x7F

Bayes=percent

Bayes reason item displays spam percentage probability computed by Bayesian Filter. It's a single value between 0 and 100 indeed.

Last possible spam reasons are BW and ContentFilter which simply signalizes that a blacklisting rule or some content filter respectively was applied for marking message as spam.

Note: MIAS logs are stored in the same location as other Merak logs and can be displayed in the Administration Console for Windows in the Logs section.

SMTP Test Tool

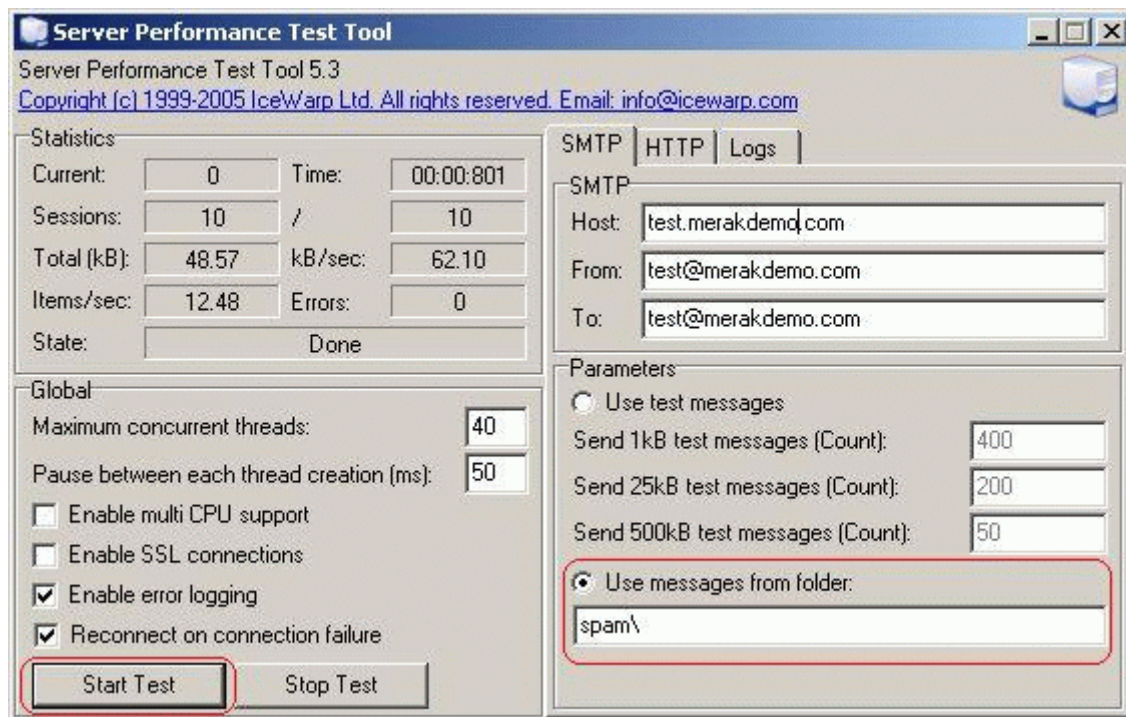
Before actual deployment of your Instant AntiSpam settings to your production server and for testing and troubleshooting, we have prepared you a smallpowerful tool - the SMTP Test Tool - that can be downloaded freely from our official website (URL: <http://www.icewarp.com/download/tools/srvtest.zip> [1.1MB]).

Download and unpack the archive. There is no need to install anything, simply run the srvtest.exe file.

The usage of this test tool is very easy. All you have to do is to enter a valid host name of the server or an IP address of the mail server. Then you have to enter a valid To email address on the server that should receive the mail. This test was mainly developed to test internal usage, but you can also test the speed of outgoing mail. For the outgoing mail test, you have to enter an external email address in the field. Pressing the START button, the test will begin. During the test, statistics will be displayed and if enabled, will also be logged.

You can also change the number of messages of each file or even take the messages from a local folder.

For Merak Instant AntiSpam purposes, you can benefit greatly from the use of the Server Test Tool. Assume you have a collection of spam messages in some sort of spam folder. To test your MIAS configuration, use the custom setting 'Use messages from folder' and point it to the folder where you previously copied your spam collection. Execute the test again and observe the results.



Now open your antispam log and check the contents (make sure you have enabled the MIAS logging). Consult the results with the previous chapter of this guide.

FAQs

Please note that these FAQs are also accessible online at URL: <http://www.icewarp.com/support/faqs/>

CHAPTER 27

Why Is A Spam Message in My Inbox Folder?

From our experience it happens very often that some users will find obvious spam messages in their Inbox. Here are the most common reasons that lead to this situation:

- § You are not using the Spam Folder at all: Spam Folder mode has to be enabled globally in the MIAS configuration and also in the user setting. (see Spam Folders)
- § MIAS is not enabled for this particular user: Check out the MIAS processing mode. If it's not set to process all, check in the account or domain configuration if AntiSpam is enabled there. (Or in Licenses dialog doubleclick on the AntiSpam module to see a list of all accounts using AntiSpam.)
- § Your licence has exceeded its limit: In Merak Administration Console, open the Licenses dialog under the Help menu and check it out.
- § MIAS engine is configured improperly: Maybe you are using too low of a score limit for marking messages as spam. Check out the MIAS log for the final spam score. That can tell you something.
- § The message was bypassed: Check the spam/spambypass.dat file for any entry that matches the sender or recipient of the message. Notice that if the message was bypassed, the Spam Assassin report is missing even if it is enabled. However, there's still an entry in MIAS log.
- § The sender is whitelisted: Check all your B&W rules that could be applied on that particular message and has 'Accept' set as filter action. Just don't forget that there are three levels of B&W Filtering (server, domain and user).
- § Authorized sender bypassing was applied: You have enabled bypassing of authorized sessions and trusted IPs and this message met such criteria.
- § Some content filter was applied: Verify that no content filter could marked this message as genuine.

CHAPTER 28

Why Is Legitimate Message in Spam Folder?

Sometimes you find a message in your Spam Folder that is a legitimate email and you don't know why it ended up there. It can be hard to discover what the exact reason was so please pay attention to the following lines for possible reasons:

- § The MIAS engine is badly configured or is just very strict and so it failed in presumption: It's recommend to first check the MIAS log for the list of spam reasons (see *Engine Logging* (on page 88)).
- § There's some blacklisting rule that marked the message as spam: Check your B&W Filters, don't forget that there are three levels of filtering - server, domain and user, and any of them can be applied.
- § Some content filter lead to a spam flag: Check all your content filters that has "Mark as spam" as a filter action.
- § You have your Challenge Response enabled and the sender didn't authorized himself: In WebMail just open the Setting - Challenge Response section, show all messages and authorize this one.

How to check the Automated General Spam Reference Base Update

The General Spam Reference Base is updated only when the version and date of the general Reference Base on your mail server differs from the one located on our servers.

If you want to test the Automated General Spam Reference Base Update, follow these steps:

- 1 Run Merak Administration Console.
- 2 Go to System -> Services and stop SMTP Service.
- 3 Open the folder C:\Program Files\Merak\spam\ and delete spam.db file.
- 4 Go to Anti-Spam menu node in Merak Administration Console
- 5 Press Update Now button.
- 6 Watch the C:\Program Files\Merak\spam\ until the spam.db file does not appear again.
- 7 Start SMTP Service again

It is good idea to use a Backup MX on high volume servers so that the mail system will continue to work during any network problems.

How to Enable SPF

If you want to know what SPF is in general, look here.

SPF feature is implemented in Merak Mail Server as a part of the SpamAssassin feature in Merak Instant Anti Spam. If you enable the SPF feature it will modify SpamAssassin score for each message.

There are three ways how the message can be evaluated against a SPF record:

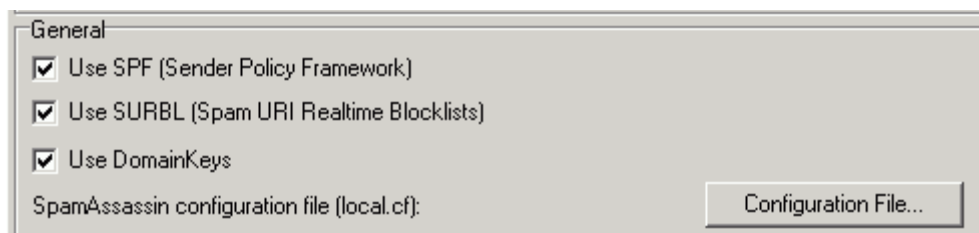
- § pass - it means that SPF record fits an IP address or hostname of sending server
- § fail - it means that SPF record does not fit an IP address or hostname of sending server
- § softfail - it means that SPF record of sending server does not exist at all so Merak Server was not able to check it

SpamAssassin score modification also depends on the sending server location (either local address or address from Internet) and if Include Bayesian probability in SpamAssassin score (bayesian in table below) option is enabled.

	Local	Internet	local + bayesian	Internet + bayesian
pass	- 0.001	- 0.001	- 0.001	- 0.001
fail	+ 5	+ 5	+ 5	+ 5.875
softfail	+ 0.5	+ 0.842	+ 0.5	+ 0.5

To enable SPF feature:

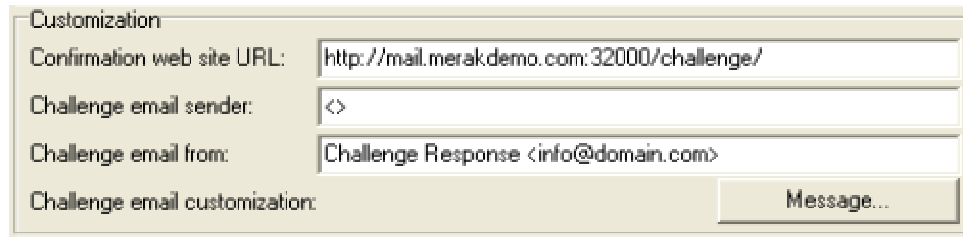
- § go to Anti-Spam menu node in Merak Administration Console
- § choose SpamAssassin tab sheet
- § enable Use SPF (Sender Policy Framework) option



How to customize Challenge Response messages?

If you are using the Challenge Response engine there is only a simple message by default. You can customize this message and also the URL on which the sender has to confirm he/she sent the message.

Go to Anti-Spam Spam -> Challenge Response tab. Below is the Customization part where you can edit the parameters.



The screenshot shows a dialog box titled "Customization" with the following fields and a button:

- Confirmation web site URL:
- Challenge email sender:
- Challenge email from:
- Challenge email customization:
- Message... button

It is only worth changing the URL in cases where you are using a multidomain configuration with a virtual host for each domain. e.g. if you have two domains: domainA.com and domainB.com and each has its virtual host webmail.domainA.com or webmail.domainB.com respectively.

You can fill in the URL field: `http://webmail.%%Recipient_Domain%%/challenge/` and in case the email is delivered to `user@domainA.com` a challenge response message will be generated with `http://webmail.domainA.com/challenge/`

Look at all the variables which are closed in %% symbols.

If you are running Control Service on different port than 80 you have to fill also the port in the URL. e.g. `http://webmail.%%Recipient_Domain%%:32000/challenge/` in case you use port 32000.

Challenge email sender parameter is used in the SMTP protocol (communication between two servers) and is recommended to remain empty because some servers can generate an auto-responder message to such addresses and a loop can arise.

Challenge email from parameter contains the address which will be shown to the recipient of Challenge Response message in the From: header. You can fill in whatever you want. It is not recommended to leave it empty because some servers reject such messages.

If you want to customize also the message itself, click the **Message** button. There you can edit challenge.txt file which contains the Challenge Response message. The syntax is:

Subject of message

<blank line>

Body of message

There is used the %s system variable. It will be substituted by the Challenge Response URL.

Example:

[Spam Challenge] Confirm your email

Visit the URL %s to confirm you email. You will be asked to do that only once and then you will be automatically authorized.

You can also use the variables such as in the URL, sender and from parameter.

How to Stop Spammers Using ESMTP and Demo/Known Accounts

Merak Mail Server installs demo accounts during its first installation. They are set to expire in 30 days from the installation but if an administrator places Merak on a production server in these 30 days or even directly installs new Merak on a production server, the server will be an open relay.

It means that spammers will be able to use admin/admin, domainadmin/domainadmin, etc. account to authenticate on Merak server and send through this server a bunch of spam messages on different servers.

Another risk results from weak passwords. For example a lot of careless administrators create accounts such as webmaster/webmaster or webadmin/webadmin as username/password.

To find out if your server is not abused as an open relay via such accounts, look at your SMTP logs.

Look for log entries like this:

```
218.27.89.158 [0000036C] Fri, 14 Mar 2003 14:00:36 +0100 <<< AUTH LOGIN
218.27.89.158 [0000036C] Fri, 14 Mar 2003 14:00:36 +0100 >>> 334 VXNlcm5hbWU6
218.27.89.158 [0000036C] Fri, 14 Mar 2003 14:00:47 +0100 <<< YWRtaW4=
218.27.89.158 [0000036C] Fri, 14 Mar 2003 14:00:47 +0100 >>> 334 UGFzc3dvcmQ6
218.27.89.158 [0000036C] Fri, 14 Mar 2003 14:00:52 +0100 <<< YWRtaW4=
```

where the AUTH LOGIN switch is using the basic way of authentication where the commands and values are Base64 encoded - it is really very simple encoding.

You can look at *free PHP encoder/decoder* (<http://makcoder.sourceforge.net/demo/base64.php>) to decode the values and you will find that the password in this example is 'admin' as well as username.

To prevent this, you should disable or change password for all demo accounts. They are in merakdemo.com domain in recent versions and only demo.com in older versions. Additionally, you should check your logs if there are not more occurrences of AUTH LOGIN and if so, find the username/password with help of Base64 encoder/decoder and disable or change password for such users.

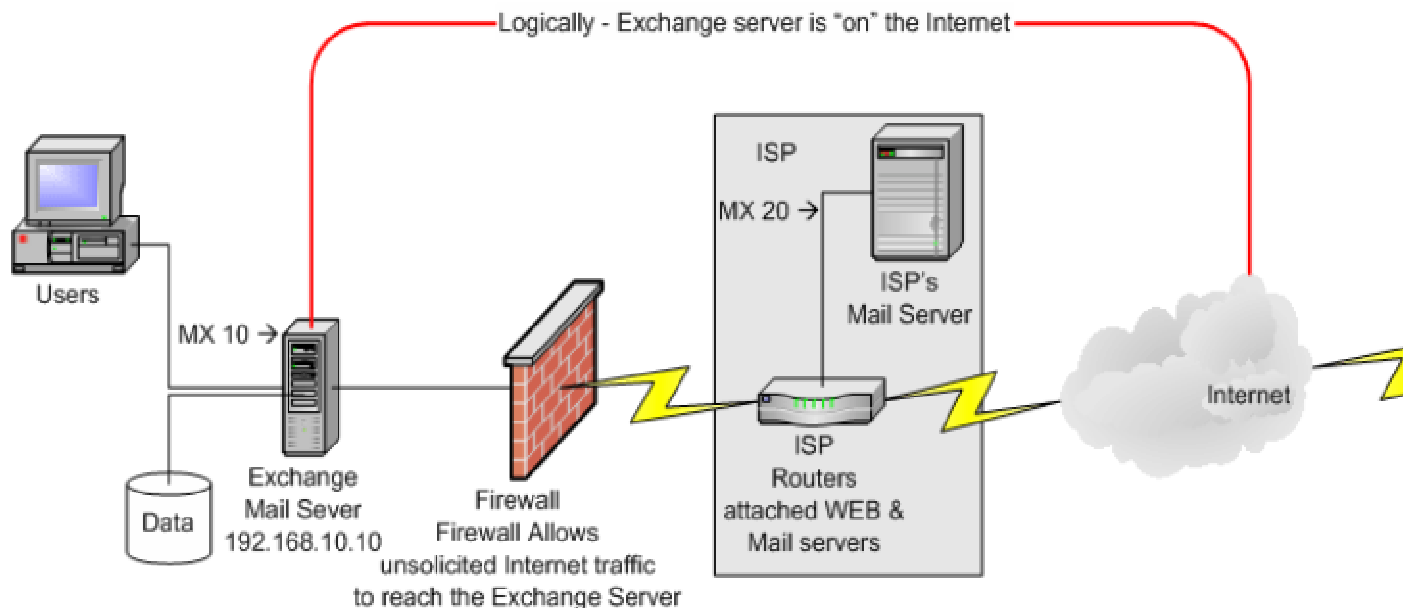
How to Use Merak as your AntiSpam and AntiVirus Gateway for MS Exchange Server?

What is it good for?

- § MS Exchange Server (hereafter only Exchange) is not directly connected to the Internet but only to your Merak Mail Server (hereafter only Merak)
- § Merak receives messages for users in domains on Exchange
- § Merak filters all messages for viruses and spam
- § Merak forwards messages to the Exchange
- § Merak offers a lot of other useful feature for SMTP processing such as Content Filters, Security features, etc.

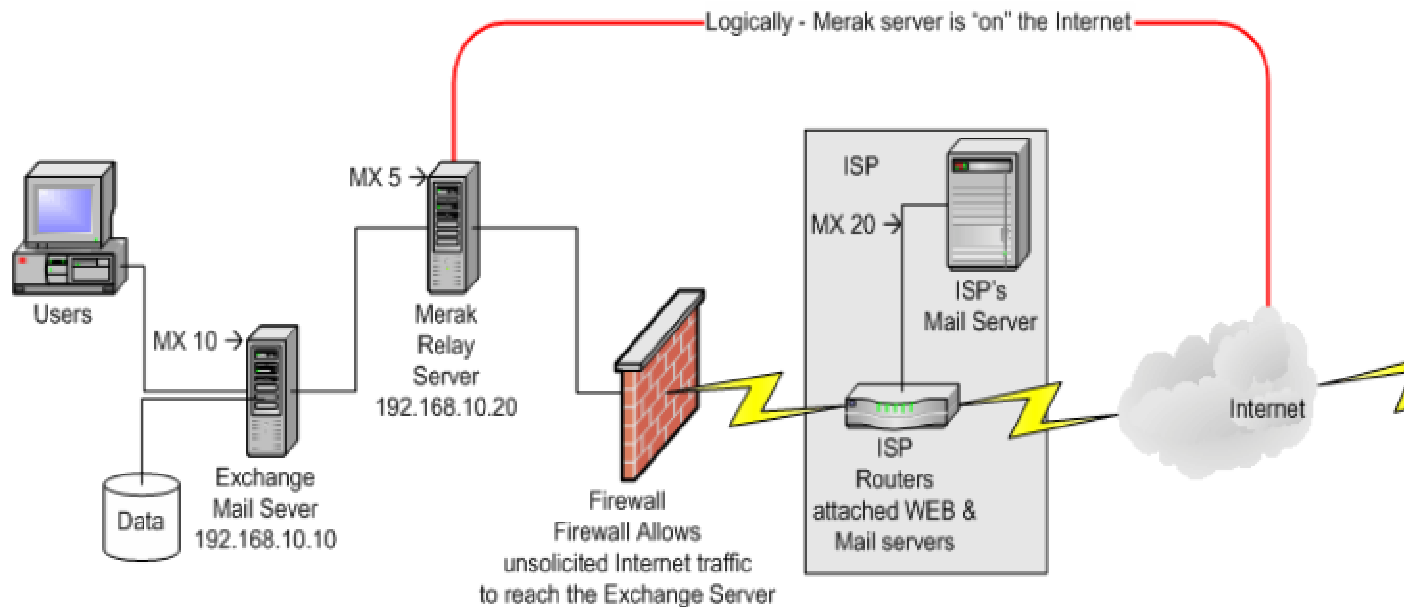
How does it work?

If you use a common configuration with Exchange as your only Mail Server it could look like on the figure below:



Source: Troup, Marty: Merak Instant Antispam as automated spam protection gateway for Microsoft Exchange; 2003; marty@icewarp.us

If you use Merak as a relay server, it means you will place it "between the Internet and Exchange". It could look like on the next figure:



Source: Troup, Marty: Merak Instant Antispam as automated spam protection gateway for Microsoft Exchange; 2003; marty@icewarp.us

All the emails are sent through Merak which can filter, forward, delete, etc. such messages according to your settings.

How to Set Merak to be Relay Server for Exchange?

1 Check Merak's DNS servers

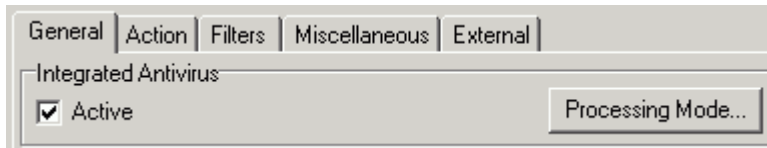
Open Merak Config go to Mail Service -> SMTP Service -> General tab and check that there is chosen Use DNS Lookup and the DNS servers are set properly. Use DNS Query Test button to verify that. Of course, set Mailserver hostname to be what you want.

General	Delivery	Redirect	Header / Footer	Other
General				
Mailserver hostname:	mail.merakdemo.com			
<input type="radio"/> Use relay server:				
<input checked="" type="radio"/> Use DNS lookup:	147.32.8.9;147.32.6.4;147.32.1.20			
DNS Query Test				

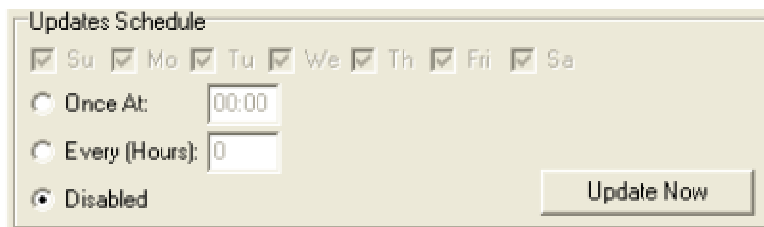
Then stop unneeded services. That is POP3/IMAP, IM (Instant Messaging) and Calendaring in System -> Services.

2 Setup AntiVirus

Go to Anti-Virus menu node. Enable Active option to enable Integrated AntiVirus.

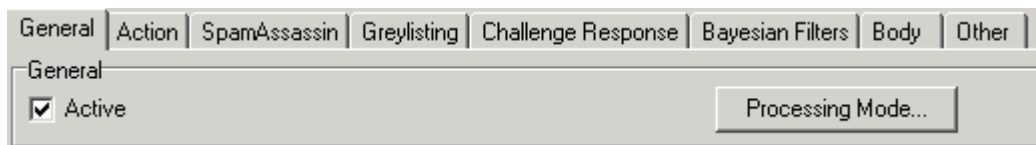


Then set Update Schedule to automatic Antivirus DB updates.



3 Setup Merak Instant AntiSpam (hereafter only MIAS)

Go to Anti-Spam menu node and enable Instant AntiSpam Engine.



You can set it according to your own needs but if you are not familiar with all the options it is recommended that you follow the settings which are set by default. To get more information regarding MIAS, you should read the MIAS documentation which is downloadable from our web pages www.icewarp.com/download/

4 Setup Different Security Features

This includes Content Filters, Tarptitting, Black & White Lists, Static Filters and many others. You can see MIAS documentation and Merak F1 help to find out more information. It would be better to tune Merak settings after some time of running Merak on your machine.

5 Setup Domains in Merak Corresponding with Domains which are Handled by Exchange

Go to Domains & Accounts -> Management and create new domain with the same name as it has on Exchange. Set domain type to Backup Domain and set the Value: option to IP or hostname of the server where your Exchange is running.

The screenshot shows a web-based configuration interface for domain management. It has a tabbed interface with tabs for 'Domain', 'Options', 'Miscellaneous', 'B&W List', and 'Info'. The 'Domain' tab is active. The form is divided into three sections: 'Domain', 'Administrator', and 'Unknown Users'.
- In the 'Domain' section, 'Name:' is 'mydomain.com', 'Description:' is empty, 'Type:' is a dropdown menu set to 'Backup domain', and 'Value:' is 'exch.mydomain.com'.
- In the 'Administrator' section, 'Default alias:' is 'postmaster;admin;administrator;supervisor;hostma:' and 'E-mail:' is empty.
- In the 'Unknown Users' section, 'Action:' is a dropdown menu set to 'Reject mail', 'E-mail:' is empty, and there is a checkbox labeled 'Send information to administrator' which is unchecked.

Click Save to save the domain settings.

Repeat this procedure for all your domains.

6 Setup DNS Records to Contain Relay Host

- § Add A record pointing the IP address of machine where Merak is running to its hostname (e.g. merak.mydomain.com to 191.191.191.11)
- § Add MX record pointing domain name to the hostname/IP of the machine where Merak is running (e.g. mydomain.com to merak.mydomain.com). Set the priority of that MX record to lower number (higher priority) than the one which is set for MX record pointing directly to Exchange.

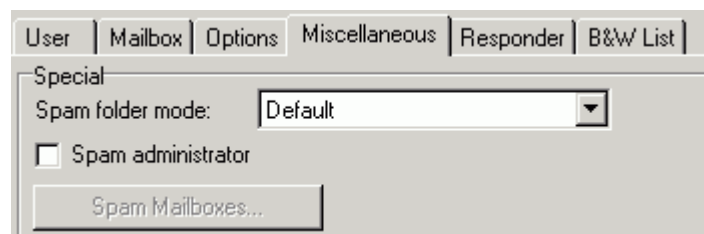
Repeat this procedure for all your domains.

How to enable spam folders for selected users

The options "Place Spam Messages Under Spam Folders" is a global setting for the entire server. However, you can override this option for each individual user in each user's settings.

To do this:

- § Go to Domains & Accounts -> Management
- § Select the user whose Spam settings you want to modify
- § Click on the Miscellaenous Tab (see below)



By using the drop down list next to Spam Folder Mode you can select:

- § Default
- § Do Not Use Spam Folder
- § Use Spam Folder (see below for details)

Option	Description
Default	The users will use the MIAS system settings. Example: Store messages to the Spam folder if the default setting for the server is set to "Place Messages Under The Spam Folder" in the Merak Instant Anti-Spam & Bayesian Engine setup.
Do Not Use Spam Folder	The Spam Folders will be not used, however, MIAS will work and the subjects will be marked with word [Spam] (if enabled).
Use Spam Folder	The Spam Folder will be used, even if it is not the default setting for the entire server.

These options provide you with maximum flexibility on a user- by-user basis. You can set some users to only receive the "[Spam]" text in their subject line (compatible with all clients), or use the "Spam Folders".

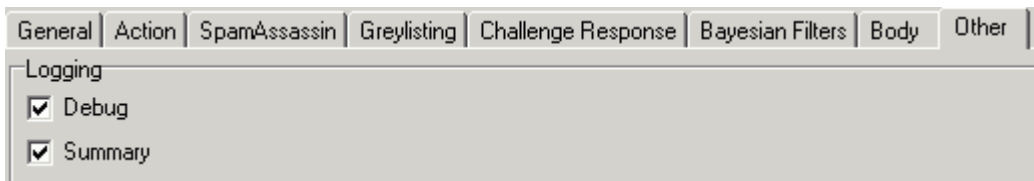
How to bypass all local messages from Spam Scanning ?

Merak Instant AntiSpam is by default being applied to all incoming as well as outgoing messages. Sometimes you need to exclude outgoing messages from spam scanning so messages from local users are not marked as spam.

So let's see how to do it. The solution has two steps.

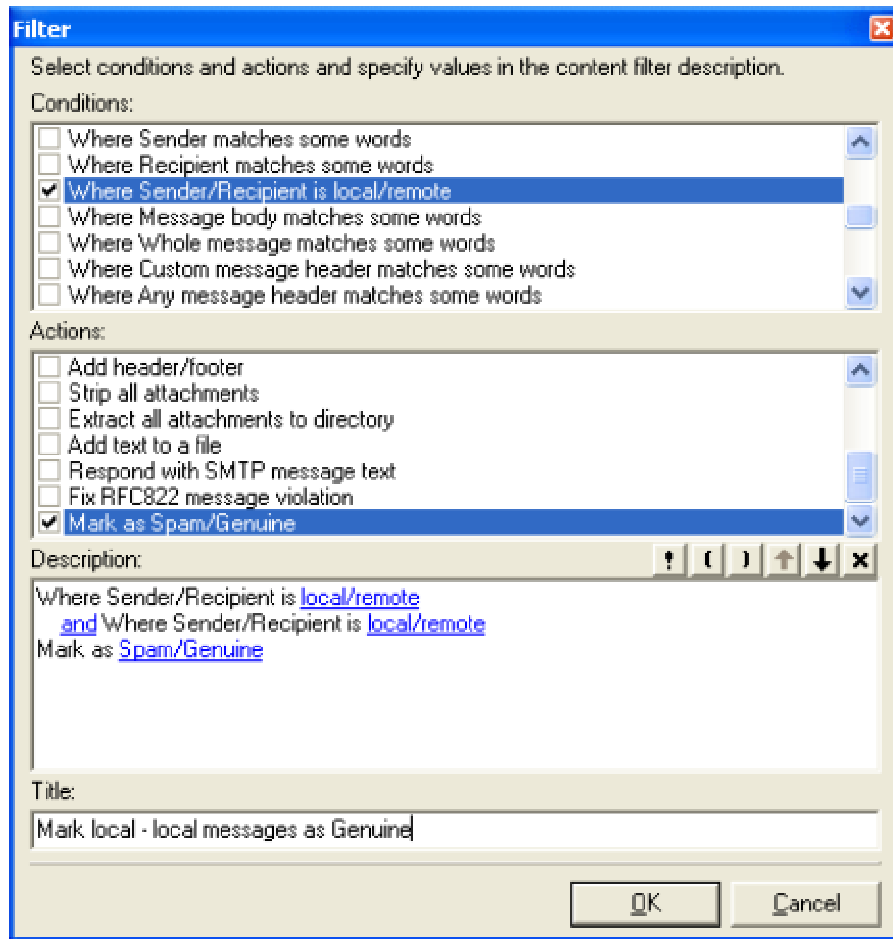
The first step:

Set Merak to bypass outgoing messages from spam scanning. Go to Anti-Spam -> Other tab and check Exclude outgoing messages from Anti Spam option.



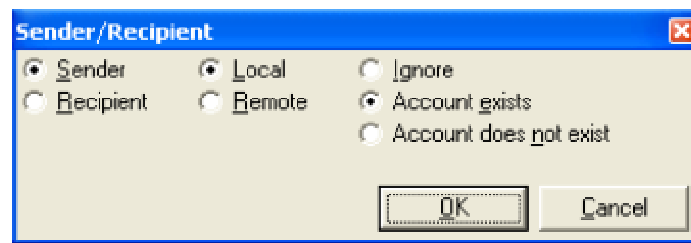
The second step:

The first step bypasses all outgoing messages. But MIAS is still being applied on all incoming messages – even the ones from local users. The solution is to set a content filter which marks all incoming messages from a local user to a local user as genuine. Go to Mail Service -> Filtering -> Content Filters -> Add and set a Content filter:

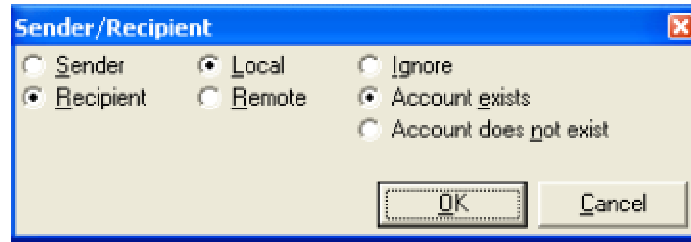


You have to click twice to a condition to add it for the second time.

The first condition is Whether Sender Is Local



The second condition is Whether Recipient Is Local



Download the filter. (local-localspam.zip)

CHAPTER 29

How to Create a Spam Trap

Many spammers use fast computers to run spam bots. A Spam bot is a program that browses the Internet. It uses hypertext links and other anchors to automatically navigate thru the World Wide Web. For every page, spam bot visits it performs a search for email addresses. These addresses are added to a database of possible recipients and then mass spamming is performed on addresses gathered in this way.

Spam trap is a mechanism, that exploits the fact that a spam bot sees text that a real user doesn't for recognizing him and ignoring all its messages. How it can be done?

If you have a website, where you want to display email addresses of some of your employers, you can hide some addresses to these pages that real user won't see. For example create a text of the same color of the page background, or better yet use CSS styles for setting this element to be hidden. This is the first step.

The second step is creating a special type of account on your Merak Mail Server. This account will have some unused email address like gotcha@domain.com. The most important step is to set the State property of this account to 'Disabled (Tarpitting)'. See the screen of Merak Administration Console below.

The screenshot displays the Merak Administration Console interface for configuring a user account. At the top, the user's name and email address are shown as 'Gotcha, <gotcha@doc.icewarp.com>', with a 'Save' button to the right. Below this is a tabbed interface with tabs for 'User', 'Mailbox', 'Options', 'Miscellaneous', 'Responder', and 'B&W List'. The 'User' tab is active, showing the following fields:

- Alias: gotcha
- Username: gotcha
- Full name: Gotcha
- Password: [masked with asterisks] Confirm: [masked with asterisks]
- Mode: Standard (dropdown menu)

A 'Comment...' button is located below the user information fields. The 'Account' section below contains the following settings:

- Type: IMAP & POP3 (dropdown menu) with an 'IMAP Rules...' button
- Permissions: Standard (dropdown menu) with a 'Rights...' button
- State: Disabled (Tarpitting) (dropdown menu)
- Forward to: [empty text field]

This sets the account to be inactive, but if email is delivered to this account, the sender is considered as a "tarpitter" and his IP address is blocked as set in the **Tarpitting** (see "Other" on page 78) options under SMTP Security section.

What will happen? The spam bot will visit your website. It loads the page with the fake address and performs a full text search in the page source code for all email addresses. To every message found it will try to send a spam message. One of these addresses is our trap. Whenever the spam bot sends a message to it, it became tarpitted and every further connection attempt is rejected immediately.

Feedback

Thank you for using Merak Mail Server.

We are always looking for ways on how to improve our services and our software. That is why, we welcome any feedback. If you have any comments or inquires about this guide or you want to share your Merak Instant AntiSpam operational strategies, please feel free to contact us by email at info@icewarp.com.

If you have problems with the software, please contact your local Merak Mail Server Support representative or use the Technical Support form at <http://www.icewarp.com/support/>.

Merak Mail and IceWarp Ltd. welcome your comments and suggestions on the quality and usefulness of this publication. Your comments and input are an important part of the information used for revision.

- § did you find any errors?
- § is the information clearly presented?
- § do you need more information? if so, where?
- § are examples correct? do you need more examples?
- § what features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the chapter, section, and page number (if available). You can send comments to us in the following ways:

- § mail to documentation@icewarp.com

If you would like a reply, please provide your name, address or SkypeID/ICQ/MSN numbers in your feedback.

Merak Mail Server Technical Writer

Limassol, Cyprus

Index

A

Administration • 12, 25, 38, 46, 56, 62, 70, 76

B

Bayesian Filters • 7, 24, 27, 52

Black and White Lists • 60, 73

Body & HTML filters • 59

C

Challenge Response • 6, 41

Configuration File • 51

Configuration Files • 7, 12, 18, 22, 23, 27, 40, 58, 66, 73, 78

Content Filters • 7, 67

D

DNSBL • 82

DomainKeys • 24, 27, 31

E

Engine Logging • 88, 96

F

FAQs • 94

Feedback • 111

G

Greylisting • 7, 35

H

How It Works • 3, 22, 23, 36, 42, 53, 60, 68, 74

How to bypass all local messages from Spam Scanning ? • 106

How to check the Automated General Spam Reference Base Update • 96

How to Create a Spam Trap • 109

How to customize Challenge Response messages? • 98

How to enable spam folders for selected users • 105

How to Enable SPF • 97

How to Stop Spammers Using ESMTP and Demo/Known Accounts • 99

How to Use Merak as your AntiSpam and AntiVirus Gateway for MS Exchange Server? • 101

I

Introduction • 1

M

Miscellaneous • 85

O

Other • 6, 78, 110

S

Sender Policy Framework and Sender Rewriting Scheme • 24, 27, 29

SMTP Test Tool • 92

Spam Assassin • 6, 21

T

Tarpitting • 79

Troubleshooting • 87

W

Why Is Legitimate Message in Spam Folder? • 96

Why Is Spam Message in My Inbox Folder? • 95