PreciseMail Evaluation Guide

PROCESS™ SOFTWARE

Overview

The first step towards eliminating spam in your organization is to choose an anti-spam filtering package. With hundreds of anti-spam solutions available, it can be a challenging task to find the one that's right for your site. Since your decision will directly affect the email of every user in your organization, it's worthwhile to carefully evaluate possible solutions. This whitepaper will provide you with what you need to determine the best solution for stopping spam at your site: several sets of criteria to look for in an anti-spam filter, objective testing procedures, and even a sample user feedback form.

Shopping for an anti-spam filter is like shopping for a new car - you want to have a basic idea of what you're looking for and what your needs are. If a particular anti-spam filter doesn't have the features you need, it's not worth taking the time to evaluate further. Some criteria you might want to look for up front are:

Criteria	Explanation
Supported platforms	An anti-spam filter needs to support your site's operating system and email server combination. Many vendors provide email proxy products that can be used with even the most esoteric messaging architectures. If your domain is served by multiple email servers, the filter should support synchronizing data between them.
User authentication	User interfaces that allow users to control filtering settings for their accounts is a must-have feature for almost every site. Most of these user interfaces require that detailed user information, especially passwords, be contained in an LDAP directory. If not all of your site's user information is contained in an LDAP directory, you need to choose a solution that supports multiple authentication methods.
Site-specific needs	If you have special site-specific needs, make sure any anti-spam filter you are considering will support those needs. For example, you may need a solution that allows you to completely customize the user interface so it is consistent with an existing email portal.
Cost	If a product costs more than you have available in your budget, then it's not worth further consideration. Note that some vendors are willing to negotiate price in return for other considerations, in addition to offering large discounts for certain customers such as educational institutions.
Technical Support	Make sure that the vendors of anti-spam filtering packages you are considering provide around-the-clock telephone support. Support by email is convenient, but by itself it's insufficient if your messaging system is inoperative.
Geographical Location	Computer software is a global market, so it's not unusual to purchase software from a vendor located thousands of miles away. If your site is located several timezones away from the vendor or you speak a

	different native language, it can cause serious communications problems. Make sure the vendor maintains offices or has a distributor in your region of the world.
Corporate stability	The demand for anti-spam filters has been increasing dramatically for the last several years. This makes it an attractive market for startup companies and individuals, who may not necessarily have the financial resources or experience to effectively support their products for the long-term. To play it safe, you should choose an anti-spam filter provided by a company with at least several years of experience with the email security market.

Criteria for Evaluation

Once you have determined which anti-spam filtering solutions fit your basic criteria, it's time to install them and see how they perform for your site. You should approach this the same way you would approach test driving a car you're interested in buying: you want to put the product through its paces and see how well it fits your particular needs.

The table below contains the major criteria you should use to evaluate each anti-spam filter. Each criterion is accompanied by detailed points to consider while evaluating the filters.

Category	Points to consider
Accuracy	• Two measures of accuracy should be used to determine which anti-spam filtering solution is right for your site: spam detection rate and false positive rate. • An anti-spam filter should have high spam detection accuracy. This accuracy number is usually expressed as the percentage of spam messages the filter correctly identified. A low spam detection rate will allow many spam messages through to user mailboxes. • An anti-spam filter should have high spam detection accuracy. This accuracy number is usually expressed as the percentage of spam messages the filter correctly identified. A low spam detection rate will allow many spam messages through to user mailboxes. • Anti-spam filters should have a very low number of false positives. False positives are legitimate messages that are incorrectly identified as spam. Even small amounts of false positives may lead to the loss or delay of important email messages. • Some anti-spam filters have a very high spam detection rate and a large number of false positives, while others have a small number of false positives and a low spam detection rate. A good anti-spam filter strikes a balance between the two extremes. • An anti-spam filter shouldn't immediately discard messages that it identifies as spam. Even the most accurate anti-spam filters available may accidentally misclassify a message. Users should be able to review and retrieve any message addressed to them that was marked as spam.

	• An anti-spam filter should be able to filter a spam message the first time it encounters it – an effective filter shouldn't require a human to recognize each individual spam attack that is occurring and write a special rule for it. Some products cannot filter spam messages until a human writes a special rule for each individual spam attack and distributes it to all systems running the filtering product. Thousands of spam messages can pass through the filter unchecked during the time it takes for the attack to be recognized, a rule to be written, and the rule to be distributed. • Rules should not be removed from the anti-spam filter after only a short period. If a rule is removed from the filter, it will not be able to recognize spam messages that have been delayed for several hours.
Configurability	• An anti-spam filter should have the ability to be tailored to fit your site's definition of spam. At the same time, a filter should be effective without requiring hours of configuration and tweaking. • System administrators should be able to control each individual rule used by the anti-spam filter to determine if a message is spam. Many anti-spam filters only allow administrators to control rules in blocks, if at all. • How "spammy" a message has to be before an action is taken should be configurable. Each user should be able to change the minimum score required to quarantine a message addressed to them. • System-wide blacklists and whitelists should be available so system administrators can block all incoming mail from known spammers. • Each user should have their own individual whitelist and blacklist that they control. If only system-wide lists are provided, they can quickly become unmanageable with the sheer volume of whitelist and blacklist entries. • The anti-spam filter's configuration and management interface should be intuitive and user-friendly. Managing the filter shouldn't require hours of the system administrator's time. • Users should not have to install additional software on their desktop systems to perform configuration tasks. Several anti-spam filters require email client plug-ins to be installed for per-user blacklist and whitelist support. These plug-ins only support a small subset of mail clients, and provide no functionality for webmail users.
Information	• An anti-spam filter is responsible for more than just deciding if a message is spam – it also needs to allow both users and administrators to see why a message was classified as spam. This information can be critical in determining if a rule should be modified or not. • A user or system administrator should be able to tell why a message was filtered simply by looking at it. Very few anti-spam filters provide information about why a message was classified as spam, and most of those require the system administrator to search through the log files looking for the message in question. Providing the information in the headers of the message itself allows a quicker turn-around time to "why was this filtered?" questions. • The product should provide succinct but useful log files. There should be one entry in a master log file for each message examined by the filter, and

	it should contain information about which rules were used to classify the message. • An anti-spam filter should provide basic statistics such as the number of messages scanned and the number of messages filtered to demonstrate the product's Return on Investment (ROI). Web-based graphical reports make it easy to see how effective an anti-spam filter is.
Methodology	• Several filtering methods are commonly used by anti-spam products. A product should use methods that allow a rich feature set while balancing accuracy and system resource consumption. • Anti-spam filtering products that only use one filtering method are easier for spammers to circumvent than products that use a mix of methods. By mixing several classes of methods, such as heuristic rule sets and Bayesian filtering, a filtering product increases its accuracy and prevents circumvention. • Avoid products that depend on easily circumvented methods such as signature checking and challenge/response. • Some products go overboard and use too many spam filtering methods. Remember that each filtering method has an associated cost in both CPU and memory usage. Products with too many methods can cause unacceptable message delays and overload a mail system.
Performance	• Email is a highly visible application to both internal and external users. Message processing delays will be quickly noticed, so an anti-spam product should not become a bottleneck. • An anti-spam filtering product should be scaleable. The volume of email sent over the Internet continues to expand, requiring constant increases in mail server capacity.
Security	• Your site's email messages are private communications that are handled by systems residing on your network. • The system administrator should have the option to control if updated spam definitions are automatically installed on the system. Automated updates are convenient and save administrator time, but some sites may have a security policy that requires them to manually inspect each new anti-spam rule or definition before it is put in place. • An anti-spam filtering product should send no information about your site to anyone without your explicit permission. Some anti-spam products send statistical information back to the company that developed them without your permission. If this information is maliciously intercepted or misused, your filtering system may be compromised. • Anti-spam proxies should support Transport Layer Security (TLS), which allows sensitive email messages to be strongly encrypted during transmission over public networks.
Time Cost	• One of the main reasons for purchasing an anti-spam filtering product is to regain time lost to spam. An effective solution should have minimal administration requirements.

	• A good anti-spam solution will allow the administrator to empower users to manage their own spam messages and filter settings. This frees up the system administrator for more important tasks, and gives end users direct and immediate control over how the spam filter deals with their messages. • The system administrator should have the option to have updated anti-spam rules automatically installed. This guarantees that new spam messages don't slip by the filter while the system administrator is busy with other tasks. • End-users should be able to preview and release their own quarantined messages. If a user has to ask a system administrator to release a quarantined message, it pulls the system administrator away from more important tasks and increases the time delay in receiving the message. • End-users should have their own whitelist and blacklist that they can manage. End-user requests to add or remove list entries at even a small organization can quickly overload a system administration team. In addition, system-wide whitelists and blacklists can quickly become too large to manage effectively if they contain entries for each user.

User Interface Evaluation

End users are constantly becoming more proficient in computer skills. As their skills increase so does their desire and ability to have more control over their personal data, including email. At the same time, rapid improvement in technology usability has created greater expectations of all user interfaces ranging from automobile dashboards and personal music players to enterprise software.

The user interface provided by spam filters should be completely customizable to fit any site's needs. It should also do well when judged by the below criteria.

Criteria	Description
Simple and Natural Dialog	The instructions and labels that appear in the interface should be written in a conversational tone. Usage of jargon or acronyms that are unfamiliar to end-users should be avoided if possible.
Natural Language Support	If the end-users of an interface speak a different language than the interface uses for instructions and labels, the interface will be virtually useless. Since most user interfaces use abbreviated language in labels, even a foreign speaker with basic fluency in the interface's language may have issues. While it is unrealistic to expect a user interface to support every conceivable language out of the box, it should provide the ability for the system administrator or a translator to rewrite all instructions and labels in the users' native language.
Minimize User Memory Load	End-users should have to remember little (if any) information specific to a given interface between usage sessions. The interface

	should be clear and intuitive, with easily obtainable help on each part of its functionality.
Consistency	A user interface should have a consistent appearance and layout with an intuitive navigation system. Changes in appearance from one area to the next can disorient and confuse users. A consistent layout also reduces the time required by new users to become comfortable with the interface.
Feedback	An interface should provide clear feedback about actions it is taking on the user's behalf. For example, if a quarantined message is released the interface should inform the user that it has been released. Simply returning the user to their home page without an informational status update can leave them in doubt as to whether the requested actions were performed or not.
Clearly Marked Exits	The user should be able to exit the interface (i.e. logout) from any place it makes sense to do so. The exit should be conveniently placed so the user can quickly logout. The user should also be able to conveniently return to their main page from any part of the interface.
Good Error Messages	When an error occurs, the interface should display an informative error message. First tier help desk staff should be able to quickly determine if a serious error has occurred based on the text of the error message. If the error message requires action from the system administrator, that action should be obvious from the text of the error.
Help and Documentation	All help and documentation required by the user interface should be self-contained. It's unreasonable to expect end-users to refer to a separate manual while inside the interface. The help should clearly describe the functionality of each feature in the interface to keep the system administrator from being bombarded with user questions.

Non-Production Evaluation Procedures

Before turning an anti-spam filter loose on your end user's mail, you might want to perform one or more of these non-production evaluation procedures. They're categorized as non-production since they don't affect end users' email in any way. (In fact, end users shouldn't even notice that you're running them.) Because they don't impact your site's actual mail, it's possible to experiment with lots of different configuration options to determine what would work best for your site.

Procedure	Description

Corpus Testing	Corpus testing is usually the first method used by sites that wish to compare the accuracy of various anti-spam filters. Large collections of messages, each of which is called a *corpus*, are sent to a test system running the filter. Counts are kept of false positives and false negatives. (See *Sources of Spam Messages* and *Sources of Non-Spam Messages* in this whitepaper for information about where to obtain messages for a corpus.) A typical corpus testing setup requires two systems, one of which is running the anti-spam filter. The other system is used to send the messages to the filter via SMTP (a simple Perl script works well for this). At the end of each test run, the anti-spam filter should be able to generate a statistical report that will show the number of messages identified as spam.
Forking User Mail	This testing method sends a copy of all messages sent to certain users to a non-production system running the anti-spam filter. This allows you to see how a filtering product performs on actual mail sent to your site, without affecting mail delivery to your user base in any way. The first step in performing this test is to select a group of volunteer users whose mail will be "forked" to the non-production system. These users shouldn't receive any mail that the IT staff isn't allowed to read (for example, the human resources director probably isn't a good test user). On the non-production system, install the anti-spam filter and set up an account for each test user. On your production MTA, create an alias for each of the test users that sends one copy of any incoming message to their "real" account, and one copy to the corresponding account on the test system. For example, if PMDF is your production MTA you would create an entry in the aliases file for each test user that looks something like: `john.doe: jdoe@example.com, jdoe@test.example.com` IT staff can log into the various user accounts on the test system to determine how the filtering solution will perform for actual users at your site.

Production Evaluation Procedures

Production evaluation procedures demonstrate how a product deals with your site's mailstream on your production email servers. Once initial testing has been conducted, these procedures can be used to determine the user population's reaction to the product.

Some basic guidelines you should consider following regardless of which production procedure you choose to implement are:

- Select a sizeable group of users to participate in the testing. The larger and more diverse a test group you assemble, the more representative their results will be of your site as a whole.
- Give the test users sufficient advance warning before the evaluation begins and ends.
- If English is not the primary language spoken by the users, customize the web interface to use the language with which they are most familiar.
- Create a mailing list for test users to which they can post problems, suggestions, and impressions. IT staff should regularly monitor the list.
- Set up aliases for test users to forward false positives and false negatives to for tracking purposes.
- At the end of the testing period, ask each test user about their email experience both before and during the evaluation period. A sample feedback form is available at the end of this document.

Procedure	Description
Quarantining	This testing method takes full advantage of the PreciseMail Anti-Spam Gateway quarantine functionality. Spam messages are placed in a quarantine area as they are received instead of being delivered to the user. Users may access their quarantined messages at any time through the web interface. Quarantine notification messages may be sent to users at administrator-determined or user-determined intervals.
Header Insertion	Special headers containing information about which rules a message triggered and whether or not it is considered spam by the filtering solution are inserted into each scanned message in this testing method. Ideally, one header line should be inserted for each data point the anti-spam filter used to make its decision, as well as a summary line giving the message's overall score. Either the end user who received the message or the system administrator can look at any message to determine why it was classified as spam or non-spam. Users who are not part of the test group will not notice anything different about their mail messages. Test users can set up rules inside their mail client to filter spam messages into a special folder based on the presence of certain headers inserted by the anti-spam filter.
Subject Tagging	This testing method places a short text string in the subject line of messages identified as spam. This allows every user to see which messages the filtering solution considers spam, without any messages being quarantined or discarded. In addition, users can set up rules inside their mail clients to filter messages with this string in the subject to a spam folder.
Log Monitoring	Every incoming mail message is scanned, but no alterations are made to messages in this testing method. The system administrator can monitor the anti-spam filter's log file and statistical reports to gauge the effectiveness of the filter. Users cannot see which messages would have been filtered as spam.

Sources of Spam Messages

One of the most popular methods for testing an anti-spam filter's accuracy is to send it a large number of spam messages and count how many get through. There are several sources of spam messages that are suitable for testing, but by far the best place to start is your own user base. Encouraging your users to save spam messages they receive to a public IMAP folder will quickly give you a corpus of spam that represents the sort of messages your users want to be filtered. (If IMAP isn't available at your site, you can have users forward their spam messages as attachments to a special account.)

If you wish to test against a broader variety of spam than is received by your user base, there are several online spam repositories. By far the largest and most popular is http://www.spamarchive.org, which averages 5,000 new spam messages a day. The spam messages are organized into compressed archives, and are freely downloadable.

Important: Spam messages from SpamArchive.org and from other online corpora usually have incomplete header information. Make sure you carefully check the messages before using them for testing, especially the Date:, To:, and From: headers. Many spam messages have incomplete or invalid headers, and anti-spam filters take that into account when deciding if a message is spam. If any of these headers are missing or invalid, simply replace them with a valid header of the same type.

Sources of Non-Spam Messages

Just like spam messages, the best source of non-spam messages to use for testing is your user base. Unfortunately, while most users are very happy to hand over a copy of every spam message they receive, they're much less likely to make even a small subset of their non-spam messages available for testing. Unlike spam messages, there are no large online repositories of non-spam messages.

One possible source of non-spam messages is NNTP newsgroups. The newsgroups provide a very large number of non-spam messages on thousands of topics. Because of the wide variety of topics, messages from newsgroups can be used to closely approximate the wide variety of email that your users receive.

Many user agents (especially Pine) are capable of pulling every message from an NNTP newsgroup and placing it in a BSD-formatted mailbox file. A simple Perl script can be used to send the contents of the BSD-formatted mailbox to a system running the anti-spam filtering solution Just like spam messages that are made publicly available for testing, you should carefully check the basic headers of messages obtained from newsgroups and replace them as needed.

Note: Spam sometimes appears in NNTP newsgroups, so an IT staff member should check the messages obtained from them and remove any spam.

Test Practices to Avoid

The following activities will produce inaccurate results while evaluating most anti-spam filtering solutions:

1. Using a small group of testers. Using a small number of user mailboxes to test an anti-spam filter will not produce a large enough quantity of spam to be statistically significant. In addition, the spam received by one user may vary widely from the spam received by other users in the organization. If possible, you should test a filtering solution against at least 50 user mailboxes to get a fair representation of how it will perform.

2. Using only testers from one department or workgroup. The content of spam and non-spam messages received by users in different departments tends to be diverse. Accuracy ratings for IT staff users may differ significantly from accuracy ratings for sales or administration staff users. Including as many diverse users as possible in the testing process provides a more accurate picture of how a filtering solution performs across the organization.

3. Forwarding spam. When a mail message is forwarded, most of the original headers are lost or modified. These headers are critical in identifying spam, so forwarding spam messages to an anti-spam filter will result in greatly reduced accuracy.

4. Using raw messages from public repositories. Messages obtained from public repositories, such as SpamArchive.org and NNTP newsgroups, will have missing or altered headers. Before you use these messages to test an anti-spam filter's accuracy, you should verify that the headers are correct and complete. If they are not, it will severely affect spam detection accuracy. Special attention should be paid to Date:, From:, and To: header fields.

5. Using homogenous message blocks. "Intelligent" filters, such as Bayesian engines, require training on both spam and non-spam messages. If you send a large block of spam messages to an intelligent filter, followed by a large block of non-spam messages, a high false positive rate will result. To more closely simulate real world conditions, you should alternate small blocks of messages to different accounts. For example, you might repeatedly send 10 spam messages to a test account, followed by 10 non-spam messages to another test account.

Sample User Feedback Form

Name: _____

Email Address: _____

Approximately how many email messages do you receive in a day? _____

Approximately how many of those messages are spam? _____

During the testing period, how many unfiltered spam messages did you receive in a day? _____

During the testing period, how many legitimate messages were filtered as spam? _____

On a scale of 1 to 5, with 1 being unlikely and 5 being likely, please rate the usefulness and ease-of-use of PreciseMail Anti-Spam Gateway:

Using PreciseMail would improve my email workflow	1	2	3	4	5
PreciseMail would keep obscene messages out of my inbox	1	2	3	4	5
PreciseMail would reduce the amount of time I spend dealing with junk email	1	2	3	4	5
I would find PreciseMail useful in my job	1	2	3	4	5
Learning to use PreciseMail would be easy for me	1	2	3	4	5
I would find it easy to get PreciseMail to do what I want it to do	1	2	3	4	5
I would find PreciseMail easy to use on a day to day basis	1	2	3	4	5
My interaction with PreciseMail would be clear and understandable	1	2	3	4	5

Comments: _____

_____

About PreciseMail Anti-Spam Gateway

PreciseMail Anti-Spam Gateway is an enterprise software solution that eliminates spam, phishing and virus threats at the Internet gateway or mail server. It has a proven 98% spam detection accuracy rate out-of-the-box without filtering legitimate messages. PreciseMail Anti-Spam Gateway has a highly sophisticated filtering engine is based on a combination of proven heuristic, DNS blacklisting, and Bayesian artificial intelligence technologies, which automatically learn how to separate spam messages from legitimate email. As a result, PreciseMail Anti-Spam Gateway can determine whether email is spam instead of passively reacting to known spammers by creating rules that block them after a spam attack occurs.

About Process Software

Process Software has been a premier supplier of communications software solutions to mission critical environments for twenty years. We were early innovators of email software and anti-spam technology. Process Software has a proven track record of success with thousands of customers, including many Global 2000 and Fortune 1000 companies.

U.S.A.: (800) 722-7770 • International: (508 879-6994 • Fax: (508) 879-0042
E-mail: info@process.com • Web: http://www.process.com/