

PureMessage for Unix

help



Contents

Getting Started.....	1
Welcome to PureMessage for Unix.....	1
Deployment Strategies.....	6
Installing PureMessage.....	18
Upgrading PureMessage.....	52
Quick Reference Guide.....	65
Contacting Sophos.....	74
Managing PureMessage.....	76
Dashboard Tab.....	76
Policy Tab.....	78
Quarantine Tab.....	128
Delayed Mails Tab.....	147
Reports Tab.....	148
Local Services Tab.....	152
Server Groups Tab.....	180
Support Tab.....	189
Administrator's Reference.....	193
PureMessage Utilities.....	193
PureMessage Services.....	195
Server Groups Management.....	198
Administrative Groups.....	201
Policy Configuration.....	242
Quarantine Administration.....	280
End User Management.....	294
Logs and Reports.....	298
Troubleshooting.....	304
Policy Script Tutorial.....	314
Policy Script Syntax Overview.....	314
PureMessage Default Policy Overview.....	323
Customizing the PureMessage Default Policy.....	328
Regular Expression Primer.....	337
About Regular Expressions.....	337
Building Simple Patterns.....	338
More Regex Resources.....	354
Copyrights and Trademarks.....	355
OpenLDAP License.....	360
SEE License.....	361
Sendmail License.....	362
Installing Sender History Database and Configuring Delay Queue.....	364
policy.siv.....	369
Glossary.....	370

1 Getting Started

This first section of the documentation is directed at administrators who are new to PureMessage. In addition to complete installation and upgrade instructions, it provides an overview of the product and its functionality, and outlines various deployment scenarios.

Related tasks

[Configuring End User Authentication](#) (page 135)

1.1 Welcome to PureMessage for Unix

PureMessage for Unix is a highly effective email filtering system that analyzes email messages at the network gateway. It protects organizations against spam and viruses and enforces corporate communication policies.

PureMessage includes a powerful set of web-based configuration and reporting tools, automated updates, and scheduled jobs that minimize the administrator's day-to-day involvement in gateway security. Administrators can manage multiple servers from a central location and, optionally, grant end users the ability to review quarantined messages themselves.

PureMessage self-adjusts its range of detection techniques as necessary to prevent protection failures. The software is automatically updated with the latest anti-virus definitions and spam rules created by [SophosLabs](#) analysts.

Automated tuning technology in PureMessage constantly balances a range of detection techniques to prevent detection failures. Sophos's Genotype technology blocks families of spam campaigns and viruses, ensuring that organizations are protected against previously unseen threats, even before specific detection is available.

If necessary, administrators can create custom rules to augment the default PureMessage policy.

PureMessage works in conjunction with one of several supported mail transfer agents. Versions of sendmail and Postfix are bundled with the product, but PureMessage also supports Oracle Communications Messaging Exchange Server, Sendmail Switch, and other versions of sendmail and Postfix.

1.1.1 How PureMessage Combats Spam

PureMessage contains thousands of spam tests that analyze individual characteristics of each message. Each of these tests has a numerical weight. When a message is analyzed by PureMessage, the weights from all the spam tests that matched the message are added up and converted to a spam score that expresses the message's "spam probability".

By default, PureMessage checks for spam in messages that originate from outside the network, but it can also be configured to test messages originating within the local domain. If a message is found to have a spam probability of 50% or more, it is copied to the PureMessage [quarantine](#). By default, PureMessage also delivers these messages, but alters the message's subject heading to show its spam probability.

PureMessage handles incoming email as follows:

1. The mail transfer agent passes a message to PureMessage.

2. PureMessage applies the policy filter to the message to test for viruses, spam indicators, or other message characteristics.
3. Depending on the results of these tests, messages are either delivered to the original recipient, redirected or quarantined.
4. PureMessage processes the quarantined messages. If quarantine digests are configured, end users can release desired messages from the quarantine. Other messages are usually automatically archived and deleted from the quarantine.

1.1.2 PureMessage Components

PureMessage is best understood as a set of components that combine to form a comprehensive system for filtering spam and viruses. While most of these components are essential to the operation and administration of PureMessage, some, such as the the End User Web Interface, Groups Web Interface and PureMessage API, are optional. PureMessage consists of the following main components:

PureMessage Manager

A web-based graphical user interface for managing and configuring PureMessage. The Manager is installed as part of each PureMessage "role" during installation and runs as the HTTPD (Manager) service. The Manager contains tabs that correspond with key areas of PureMessage functionality (for example, Policy and Reports). PureMessage features are also accessible from the command line. By default, these commands are run as the 'pmx6' user.

PureMessage Services

PureMessage operates as a series of services. Background services such as HTTPD (Manager) and Milter (Policy) are activated when PureMessage starts. PureMessage also uses a set of scheduled jobs to perform a variety of administrative tasks at specified times. These jobs are controlled collectively by the "Scheduler", which itself is a Background Service. The status of all enabled services is displayed on the Local Services tab of the PureMessage Manager.

Policy Engine

A powerful configuration and management tool for message filtering, the PureMessage policy engine uses a set of rules to process messages that pass through the PureMessage mail filter. The [Sieve](#)-based policy engine runs as the Milter (Policy) service. It can be managed and configured from either the PureMessage Manager or the command line. Policy lists and maps can also be configured so that conditions and actions are applied to the email addresses or hostnames contained in a particular list or map.

Quarantine

The PureMessage quarantine holds messages that have been quarantined according to the rules applied by the PureMessage policy engine. Quarantined messages are managed using either the `pmx-qman` command-line program or via the Quarantine tab in the PureMessage Manager. Depending on PureMessage installation options, end users can manage their own quarantined messages using the End User Web Interface (EUWI).

Reports

PureMessage generates pre-defined reports that provide graphical or tabular data on key performance statistics. To use these reporting features, the [PostgreSQL](#) database must be enabled. In multi-server configurations, data is collected from various network locations and is stored in a centralized database.

End User Web Interface

A web-based interface that allows end users to manage their own quarantined messages. The amount of control granted to users depends on the settings specified by the PureMessage administrator. The EUWI allows users to perform such tasks as approving blocked messages, deleting messages, and creating lists of approved and blocked senders. The EUWI runs as a PureMessage service and, by default, uses port 28443.

Groups Web Interface

A web-based interface that allows a global administrator to delegate administrative responsibilities to "group" administrators based on groups/domains and/or roles. Delegated tasks can include quarantine management, reporting, list management and the configuration of certain policy settings. Even if you do not plan to deploy PureMessage under the groups model, you can use the quarantine and reporting features as an alternative to equivalent features in the PureMessage Manager.

PureMessage API

The PerlMx module specifies the PureMessage interface for writing sendmail filter modules in Perl. In addition, it includes the "glue" code required to register a new filter and run the filter as a standalone process.

1.1.3 Using the PureMessage Documentation

The PureMessage documentation is divided into four main sections: *Getting Started Guide*, *Manager Reference*, *Administrator's Reference*, *Appendices*.

If you are new to PureMessage, it is recommended that you review the following sections of the documentation first:

- Read the [Quick Reference Guide](#) to gain a basic understanding of the PureMessage components and their functionality.
- Review the [Deployment Strategies](#) section to help you decide the best way to configure and test PureMessage in your environment.
- See [Installing PureMessage](#) for instructions on using the PureMessage installer and various post-installation procedures.

Complete PureMessage documentation is available via the PureMessage Manager.

Note

If you launch the documentation from the Manager, the documentation you are viewing is for your specific version of PureMessage.

Documentation is also available at the command line (explained below) and in PDF format. Click the desired link at the bottom of the table of contents pane of the online help to view/print a PDF in either US Letter or A4 format.

There are two ways to access PureMessage documentation from within the PureMessage Manager. The first is to click *PureMessage User Guide* on the **Support** tab. The second is to click the **Help** link that appears on the sidebar of each Manager page. Additional links on these pages provide access or other Manager Reference topics, and related information elsewhere in the documentation or on the internet.

The PureMessage documentation consists of the following main sections:

Getting Started Guide

This first section of the documentation is directed at administrators who are new to PureMessage. In addition to complete installation and upgrade instructions, it provides an overview of the product and its functionality, and outlines various deployment scenarios.

Manager Reference

The *Manager Reference* describes all of the essential administrative tools for configuring and operating PureMessage via its web-based interface. On occasion, if configuration by way of the Manager is not possible, command-line instructions are provided. This documentation begins with a section on Using the PureMessage Manager and is then organized according to the tabs in the PureMessage Manager user interface. The Manager tabs are:

- Dashboard Tab
- Policy Tab
- Quarantine Tab
- Reports Tab

- Local Services Tab
- Server Groups Tab
- Support Tab

Administrator's Reference

The *Administrator's Reference* contains detailed information about key areas of PureMessage functionality such as the PureMessage Policy, the PureMessage Quarantine and PureMessage Services. The *Administrator's Reference* also contains links to the PureMessage pages. See the relevant section (for example, "Server Groups Management") to view a list of links to related man pages.

Appendices

The *Appendices* include tutorials and links to third-party documentation. In addition to general reference documentation, such as licensing information and Frequently Asked Questions, the appendices of the *Administrator's Reference* contain frequently asked questions and a glossary of terms.

Command-Line Documentation

If you prefer to manage PureMessage via the command line, the documentation consists of man pages for each of the command-line programs as well as descriptions of the options available in the various PureMessage configuration files.

While logged in as the PureMessage user (by default, 'pmx6'), enter `man <program_name>`. For example, enter `man pmx-license` to display the licensing documentation.

Related concepts

[Quick Reference Guide](#) (page 65)

[Deployment Strategies](#) (page 6)

[Installing PureMessage](#) (page 18)

[Administrator's Reference](#) (page 193)

The Administrator's Reference provides an in-depth examination of PureMessage. PureMessage began as a Unix-based, command-line mail filtering tool, and it still retains many aspects of its origins.

[Managing PureMessage](#) (page 76)

The PureMessage Manager is a web-based graphical interface to PureMessage. Although there are some administrative tasks that can only be performed from the command line, the majority of tasks can be accomplished via the Manager.

1.1.4 Documentation for Related Applications

Sendmail Documentation

If you are using sendmail or Sendmail Switch as your mail transfer agent, refer to the documentation included in your installation. If you have installed the version of sendmail distributed with PureMessage, the sendmail documentation is located in `/opt/pmx6/sendmail/doc`.

- `/opt/pmx6/sendmail/doc/op.txt` : Sendmail Installation and Operation Guide
- `/opt/pmx6/sendmail/doc/op.pdf` : Sendmail Installation and Operation Guide
- `/opt/pmx6/sendmail/doc/FAQ` : Sendmail FAQ
- `/opt/pmx6/sendmail/doc/KNOWNBUGS` : Known Bugs in Sendmail
- `/opt/pmx6/sendmail/doc/LICENSE` : Sendmail License
- `/opt/pmx6/sendmail/doc/README` : Sendmail Readme
- `/opt/pmx6/sendmail/doc/RELEASE_NOTES` : Sendmail Release Notes

Some sendmail documentation is available on the sendmail website.

Postfix Documentation

If you are using Postfix as your mail transfer agent, refer to the documentation included in your installation. If you have installed the version of Postfix distributed with PureMessage, the Postfix documentation is located in `/opt/pmx6/postfix/doc`.

- `/opt/pmx6/postfix/doc/index.html` : Postfix Introduction
- `/opt/pmx6/postfix/doc/basic.html` : Postfix Configuration
- `/opt/pmx6/postfix/doc/motivation.html` : Postfix Overview

- `/opt/pmx6/postfix/doc/receiving.html` : Postfix Anatomy
- `/opt/pmx6/postfix/doc/faq.html` : Postfix FAQ

Postfix documentation is also available on the [Postfix](#) website.

Oracle Communications Messaging Exchange Server Documentation

If you are using Oracle Communications Messaging Exchange Server, you can configure a direct connection with the PureMessage milter as described in “Configuring PureMessage for Oracle Communications Messaging Exchange Server” in the Sophos Knowledgebase. For further information, see the Oracle documentation.

PostgreSQL Documentation

If you are using PostgreSQL to consolidate quarantined messages, refer to the documentation on the PostgreSQL website: <http://www.postgresql.org/docs/>.

Related information

[Configuring PureMessage for Oracle Communications Messaging Exchange Server](#)

1.2 Deployment Strategies

This section outlines possible implementation and server deployment scenarios, describes PureMessage configuration and testing, and suggests methods for ongoing tuning and optimization.

1.2.1 Deploying PureMessage

This document outlines possible implementation and server deployment scenarios.

Post-Installation Testing

After installing PureMessage, you can test and validate many PureMessage functions without interrupting mail flow. Testing options are described in the sections that follow. See “Testing PureMessage Operations” in the Installation Guide for more information.

Stages of Deployment

In most cases, PureMessage is deployed in four stages:

- Stage 1: Server Setup
- Stage 2: Message Handling
- Stage 3: Collect and Analyze User Feedback
- Stage 4: Tuning

The initial stage does not route any live mail through PureMessage, which allows for offline testing and validation of the installation. The next stage routes live mail through PureMessage, but does not prevent the delivery of any messages (thus preventing disruption of mail delivery but allowing testing with live messages for the purpose of verifying policy configuration). The latter stages involve quarantining messages, implementing quarantine digests for user notification and ongoing tuning.

Related tasks

[Testing PureMessage Operations](#) (page 41)

Testing with a Sub-Group of Users

This scenario describes enlisting a small group of users to test PureMessage with live traffic. The default PureMessage policy does not prevent any messages from being delivered. Therefore, this stage of testing does not disrupt the mail flow.

- **Server Setup:** Create a test domain to run a live test. Route a select group of real user mailboxes to the test server.
- **Message Handling:** By default, the PureMessage policy adds a custom message header to all messages with a spam probability. This message header also records the anti-spam rules that were triggered by the message. Messages with a probability greater than 50% are copied to the quarantine, in addition to being delivered to the original recipient. See “Message Handling Options” in the Quick Reference Guide for more information.
- **User Feedback:** Create mailboxes for collecting mis-classified messages. Analyze the messages submitted to these mailboxes. See “Gathering User Feedback” for more information.
- **Tuning:** Analyze messages forwarded to the mailboxes, tune PureMessage by using [whitelists](#) and [blacklists](#) to exempt messages from tests.

Related concepts

[Message-Handling Options](#) (page 70)

[Gathering User Feedback](#) (page 13)

[Tuning Spam Detection](#) (page 14)

Testing with a Full Domain

This scenario describes testing with a full, live domain, which allows you to analyze filtering effectiveness and system throughput. As in the previous scenario, all messages are delivered, so mail traffic is not disrupted.

- **Server Setup:** Use a live server to relay live traffic to the full mail domain.
- **Message Handling:** Modify the default policy script so the subject line of the original message is not altered. See “Message Handling Options” in the Quick Reference Guide for more information. By default, messages with a spam probability over 50% are copied to the quarantine.
- **User Feedback:** Analyze the quarantined messages to determine common characteristics that can be addressed through tuning.
- **Tuning:** Tune by using [whitelists](#) and [blacklists](#) to exempt messages from tests and reduce false positives.

Related concepts

[Message-Handling Options](#) (page 70)

[Gathering User Feedback](#) (page 13)

[Tuning Spam Detection](#) (page 14)

Deployment

In a fully deployed PureMessage installation, end users do not receive any quarantined messages. Rather, PureMessage generates Quarantine Digests that list the quarantined messages. Users can release messages from the quarantine by clicking on the desired message. See “Quarantine Digests” in the Quarantine Management section of the *Administrator's Reference* for more information.

- **Server Setup:** Use a live server to relay live traffic to the full mail domain.

- **Message Handling:** Modify the policy script to quarantine messages above a certain spam threshold.
- **User Feedback:** Implement Quarantine Digests so users can see which messages have been quarantined and release the messages they want to receive. As described in the “Message Handling Options” section of the Quick Reference Guide, users can submit mis-classified messages to the "is-spam" and "not-spam" accounts.
- **Tuning:** On an ongoing basis, tune spam detection based on user feedback and quarantine analysis.

Related concepts

[Digests Management](#) (page 286)

[Message-Handling Options](#) (page 70)

[Tuning Spam Detection](#) (page 14)

Server Deployment Options

PureMessage is a collection of components bundled into [roles](#). These roles can run on a single server or be shared across multiple servers in various configurations. This section reviews various deployment scenarios.

This section suggests three PureMessage server deployment options ranging from simple implementations (for smaller organizations) to more complex deployments for organizations with higher volumes of mail. PureMessage scales both vertically (bigger servers, faster processors) and horizontally (more servers). The PostgreSQL based back-end quarantines have been designed to scale to tens of millions of messages. Server management is scalable by using the PureMessage Server Groups features to replicate configuration and aggregate reporting across multiple servers.

Different levels of performance can be achieved by installing PureMessage roles on either a single server, or in varying configurations on multiple servers.

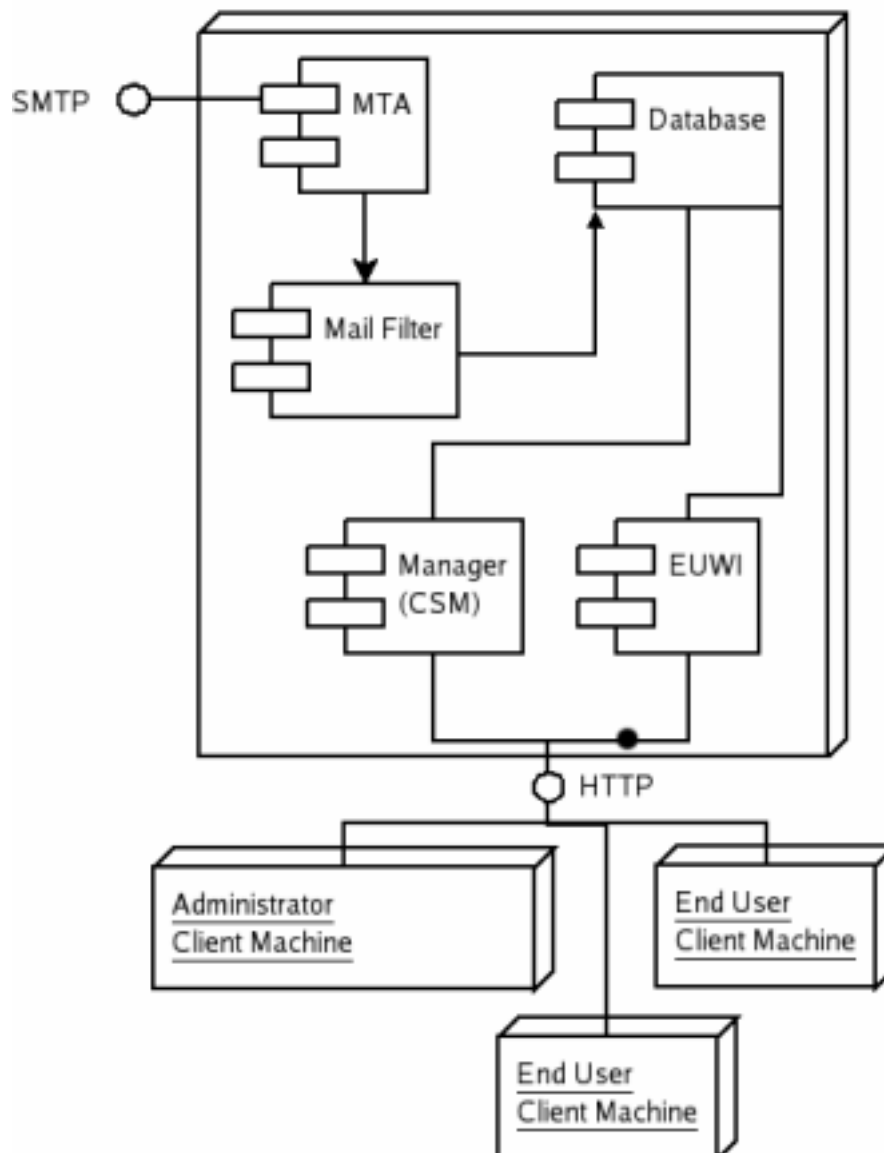
Related concepts

[Server Groups Tab](#) (page 180)

Single-Server Deployment

A single-server deployment is recommended for organizations with lower mail volumes. In this configuration, a single PureMessage server includes all of the roles that can be deployed to a PureMessage server.

Single-Server Deployment

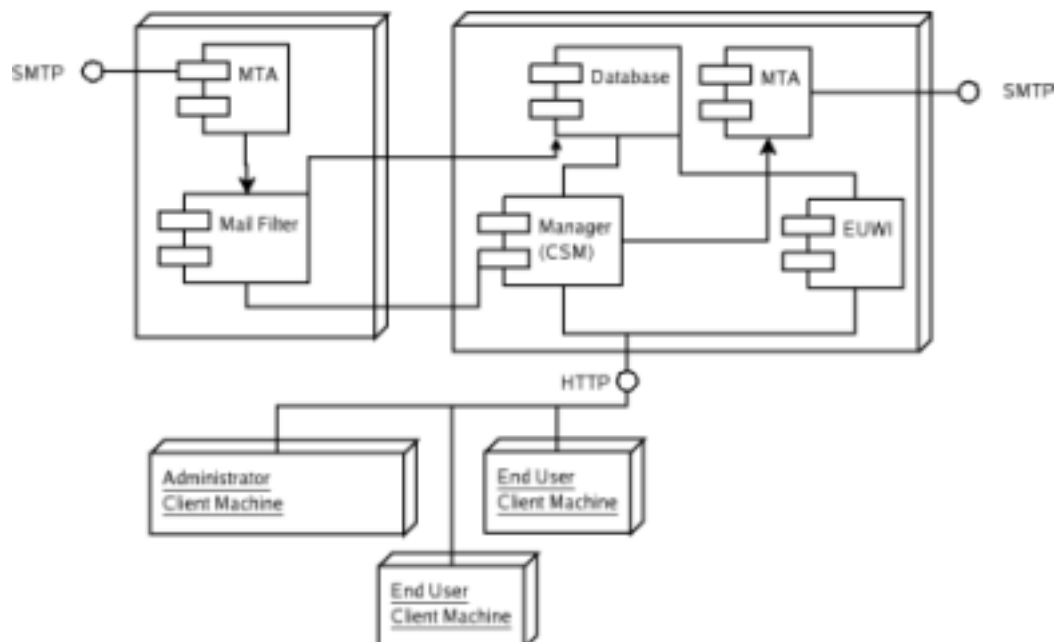


This default "Full" installation is well suited to small organizations. For higher volumes of mail traffic or a very large quarantine you should consider a deployment with two or more servers.

[Multi-Server Deployment: Two Servers](#)

This multi-server deployment deploys Mail Transfer Agent and Mail Filter roles on a separate server. A server that assumes these two roles is called an "edge" server. The other server in this deployment assumes the three remaining roles.

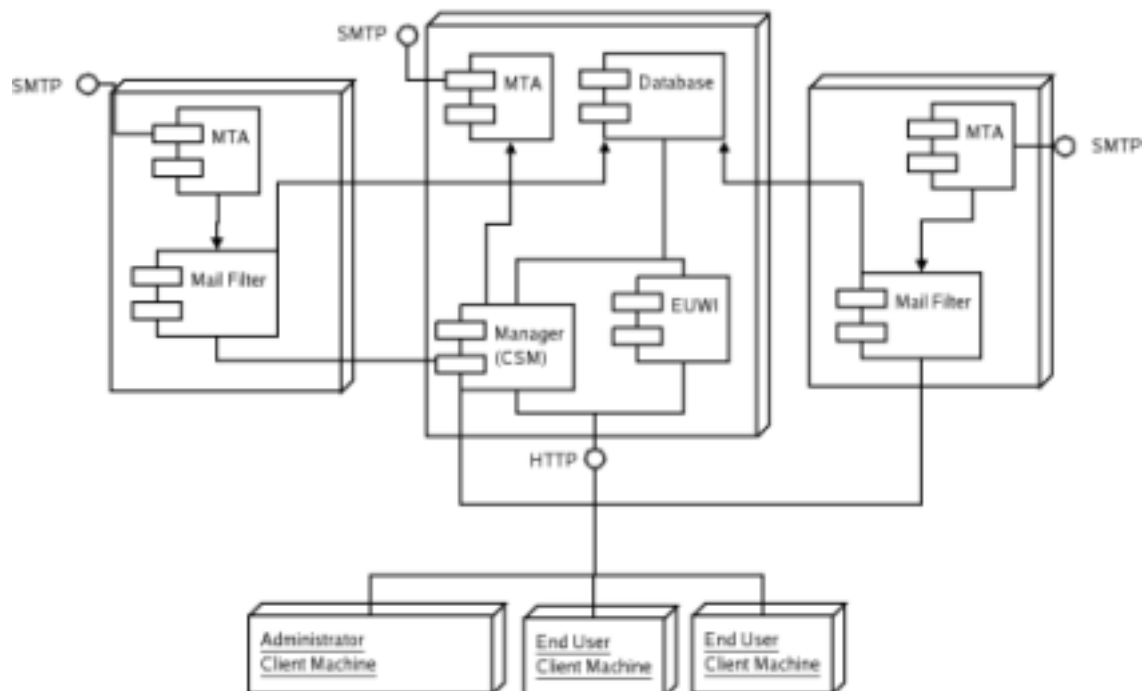
Multi Server Deployment 1



Multi-Server Deployment: Three Servers

This deployment scenario applies to organizations using multiple Mail Transfer Agents (MTAs) and PureMessage servers to handle the volume of mail associated with a larger user base.

Multi Server Deployment 2



This deployment is similar to the previous example in that a single server runs the Centralized Server Manager, Database Server and End User Web Interface Server roles. In this configuration, multiple [edge servers](#) handle a higher volume of mail traffic and filtering.

You can scale PureMessage by adding more edge servers for increased capacity.

Configuring Optional Components

The components described in this section require manual configuration. See the individual topics for descriptions and further instructions.

Quarantine Digests

Quarantine digests are lists of quarantined messages sent to the users for whom the messages were originally destined. Users can release messages from the quarantine by replying to the digest. Quarantine digests are not enabled by default; they must be manually configured and enabled. See “Quarantine Digests” in the Quarantine Management section of the *Administrators’s Reference* for instructions on configuring and generating quarantine digests.

Related concepts

[Digests Management](#) (page 286)

Centralized or Consolidated Quarantine

PureMessage has two options for managing quarantined messages: centralized or consolidated. In a centralized quarantine, metadata from multiple quarantines is collected in a single PostgreSQL database. In a consolidated quarantine, messages are drawn from multiple quarantines and stored in a single location. For more information, see “Consolidated vs. Centralized Quarantines” in the Quarantine Management section of the *Administrator’s Reference*.

Related concepts

[Consolidated vs. Centralized Quarantines](#) (page 284)

Server Groups

Server groups consist of multiple servers that are managed, configured, and maintained from one central server. Server Groups provide the ability to update configuration files, consolidate reports from multiple servers, and view, start and stop services running on other PureMessage servers. See “Server Groups Tab” in the *Manager Reference* for instructions on creating and administering server groups.

Related concepts

[Server Groups Tab](#) (page 180)

Perimeter Protection

Perimeter protection analyzes log activity and generates reports when configured benchmarks, such as the number of recipients or the maximum message size, are exceeded within the specified time frame. Perimeter protection is not configured to run by default and must be manually configured and enabled. “See Setting Log Watch Options” in the Local Services section of the *Manager Reference* for instructions on configuring perimeter protection.

Related concepts

[Server Groups Tab](#) (page 180)

End User Web Interface

The End User Web Interface (EUWI) provides individual user access to PureMessage mail-filtering options. These include viewing and managing quarantined messages, managing user-specific [whitelists](#) and [blacklists](#). End users also have the ability to configure mail-filtering options.

The default installation automatically grants all PureMessage end users permission to use the EUWI. This is done by adding them to the **End Users** list using the **Policy** tab of the PureMessage Manager. All end user options and mail-filtering functionality are available by default.

Note

Depending on your implementation, you may want to limit EUWI access to specific users or limit the options available to them. See “Configuring End User Features” in the Quarantine section of the *Manager Reference* for more information.

The EUWI is comprised of the following components:

- **Navigation Menu:** The menu located on the left side of the page.
- **Main Page:** The current page accessed through the navigation menu. Links to additional pages, including “Blocked Messages”, “Deleted Messages”, “Approved Senders”, “Blocked Senders”, and “Options”.
- **Email Message Viewer:** A viewer that opens when the “Subject” link in a message is clicked.

The EUWI uses different, user-friendly names to describe the same functionality available in the PureMessage Manager.

- The Manager’s whitelisted senders and hosts are called “Approved Senders” in the EUWI.
- The Manager’s blacklisted senders and hosts are called “Blocked Senders” in the EUWI.
- The Manager’s quarantine is called “Blocked Messages” in the EUWI.

To access the EUWI, users must log on to the web server on the EUWI host via port 28443. Their identities must be authenticated. There are three authentication methods:

- **Email Session Authentication:** This is the default method to authenticate end users. During the initial login, users are prompted for their email address and they request a password. A session ID key is automatically generated and emailed to them as the password for their access.
- **Password File Authentication:** This optional authentication method requires that the PureMessage administrator add user names and initial passwords to the `enduser/enduser_ui_user_passwords` file and email the initial password to each end user. Encryption can be applied to the password file, which is advised.
- **LDAP Authentication:** This optional method uses an existing LDAP server to perform the user authentication. PureMessage currently supports Active Directory, Sun ONE Directory Server 5.2, and OpenLDAP.

Configure the authentication method on the **End User Authentication** page of the PureMessage Manager’s **Quarantine** tab. For more detailed explanations of the authentication methods, see “Authenticating End User Access” in the End User Management section of the *Administrator’s Reference*.

Use the Manager to configure the options available to users in the EUWI. See “Setting End User Options” in the Quarantine section of the Manager Reference to set the location and session options for the EUWI. See “Configuring End User Features” in the Quarantine section of the *Manager Reference* to set the EUWI features that are accessible to end users. See “Managing End User Whitelists” and “Managing End User Blacklists” in the Quarantine section of the Manager Reference to manage whitelists and blacklists for individual end users.

The EUWI is implemented as a service that can be managed from the **Local Services** page of the PureMessage Manager. This service synchronizes per-user list changes (for example, whitelists and blacklists) to all PureMessage hosts.

Related concepts

[Policy Tab](#) (page 78)

[Quarantine Tab](#) (page 128)

[About End User Authorization Methods](#) (page 138)

Related tasks

[Configuring End User Features](#) (page 134)

[Configuring End User Authentication](#) (page 135)

[Managing Scheduled Jobs](#) (page 172)

Groups Web Interface

The PureMessage Groups Web Interface allows a global administrator to delegate administrative responsibilities to "group" administrators based on groups/domains and/or roles. Delegated tasks can include quarantine management, reporting, list management and the configuration of certain policy settings. Group administrators can only access tabs and features that have been made available by the global administrator.

The Groups Web Interface is a PureMessage service that runs, by default, on port 28443. Group Administrators can only access the tabs and features that have been made available by the global administrator. For more information, see the "Groups" overview.

Related concepts

[Administrative Groups](#) (page 201)

This section describes the setup and Management of the Groups Web Interface, which a system administrator can use to delegate selected tasks to other system administrators.

Gathering User Feedback

User feedback is gathered to determine the effectiveness of the policy configuration, in particular the accuracy of spam detection. See the "Configuration and Tuning" section of this guide. for information about applying changes based on user feedback. User feedback options vary according to the implementation scenario and stage of deployment. For example, if you have not yet implemented Quarantine Digests, that aspect of user feedback does not apply.

- **Reporting Mailboxes:** Create a mailbox called "is-spam" and another called "not-spam". Instruct users to forward mis-classified messages to these mailboxes for the purpose of analysis and anti-spam rule tuning. For example, when users receive a message they think should have been classified as spam, they should forward the message to the "is-spam" mailbox. Conversely, when users receive a quarantine digest that contains a legitimate message, they should release the message from the quarantine, and then forward the message to the "not-spam" mailbox. (To preserve the original message headers, instruct users to forward messages as attachments, rather than using their mail client's "Forward" function.)
- **Quarantine Analysis:** Use the **Quarantine** tab of the PureMessage Manager to analyze messages that have been identified as spam. This analysis can be used to exempt legitimate senders from spam scanning via [whitelisting](#), to automatically quarantine illegitimate senders via [blacklisting](#), and to determine common spam characteristics that can be addressed via anti-spam rule modifications.

Related concepts

[Configuration and Tuning](#) (page 13)

[Server Deployment Options](#) (page 8)

1.2.2 Configuration and Tuning

While tuning is generally an ongoing task, the most intensive period of tuning occurs during the early implementation stages, when you are customizing PureMessage to suit your organization. Much of the burden of ongoing tuning is reduced by the automated updates to the anti-spam and anti-virus [heuristics](#). These updates are run as scheduled jobs and are enabled by default. Therefore, in most

cases it is only necessary to tune your installation to customize it to your environment, rather than to address emerging spam techniques.

Tuning Spam Detection

Optimizing spam detection requires finding the right balance of false negatives (spam messages incorrectly classified as legitimate mail) and false positives (legitimate messages incorrectly classified as spam). While most customers achieve an acceptable level of spam filtering from the default PureMessage configuration, ongoing tuning can optimize the spam catch-rate and minimize false positives.

PureMessage includes a large set of anti-spam rules used to identify message characteristics that indicate spam. Each rule has a positive or negative weight. When the policy script executes the "Spam probability" test, the set of anti-spam rules is tested against the message. When a message triggers an anti-spam rule, the weight is added to the message; the combined weights form the total score, which is converted to a percentage.

Within the PureMessage policy, actions are associated with a message's spam probability. These "thresholds" perform different actions based on the likelihood that the message is spam. For example, the default policy copies messages with a probability of 50% or greater to the quarantine; at the 20% threshold, the policy adds a custom header. Multiple thresholds and associated actions can be defined within your policy.

Tuning PureMessage can involve changing anti-spam rule weights, adding custom anti-spam rules, modifying policy rule thresholds, and populating lists for use by the policy (for example, whitelists and blacklists). These processes are described below.

Whitelisting

"Whitelists" are lists of senders and/or domains that are known to be legitimate sources of email. By default, email from whitelisted hosts and senders is delivered without being scanned for spam. See "Lists" in the Policy Configuration section of the *Administrator's Reference* for information about configuring whitelisted hosts and senders.

Blacklisting

"Blacklists" are lists of senders and/or domains that are known to be sources of illegitimate email. By default, email from blacklisted hosts and senders is quarantined without being scanned for spam. See "Lists" in the Policy Configuration section of the *Administrator's Reference* for information about configuring whitelisted hosts and senders.

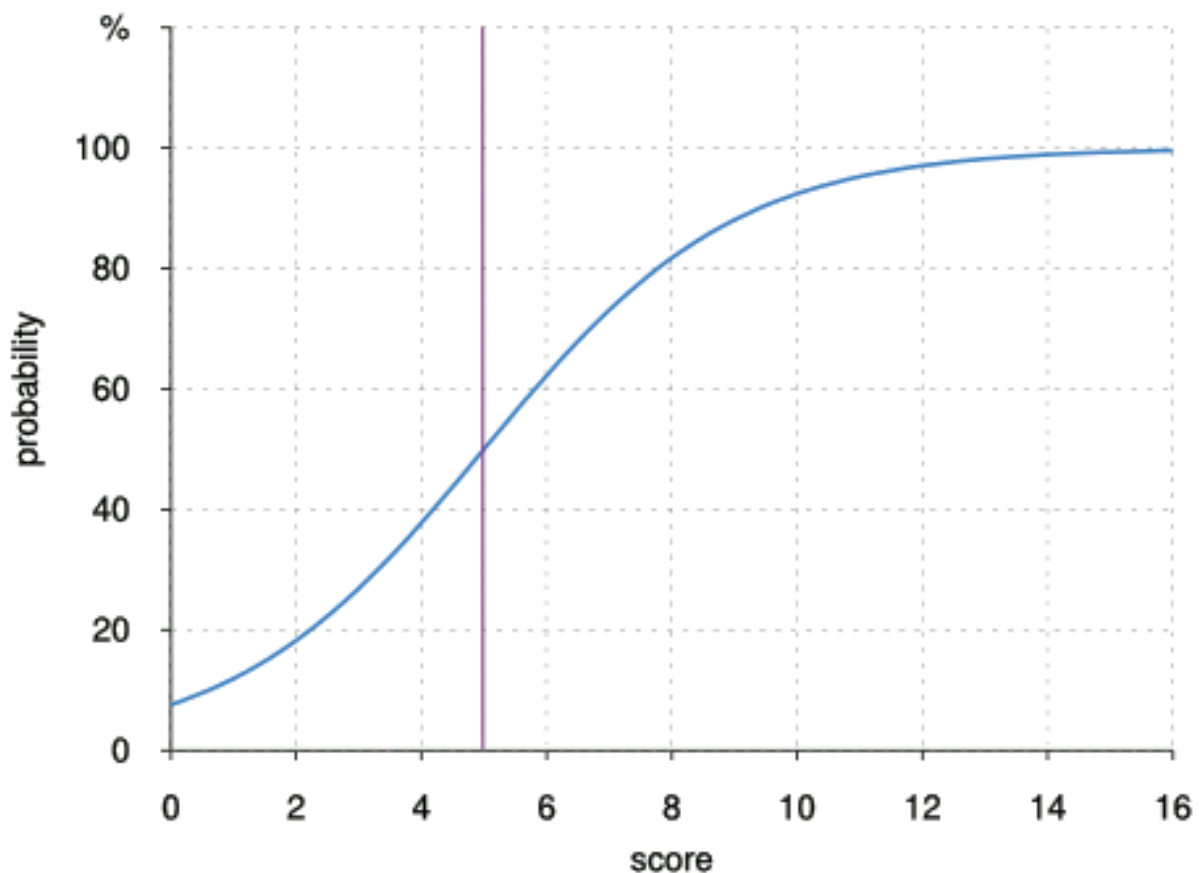
Altering Anti-Spam Rules

PureMessage includes a large set of anti-spam rules designed to detect spam characteristics.

- **Changing Rule Weights and Probabilities:** Only the weight and probability of default anti-spam rules can be changed. Rules can be viewed on the **Anti-Spam Rules** page of the PureMessage Manager, or using the `pmx-spam` command-line program. Existing rules can be disabled, or the weights can be altered. See "Adding and Configuring Rules" in the Policy Configuration section of the *Administrator's Reference* for more information.
- **Creating Custom ("Site") Anti-Spam Rules:** Custom rules that use regular expressions to test for spam characteristics can be added on the **Anti-Spam Rules** page of the PureMessage Manager, or edit the site-specific anti-spam rule files. See `re.rules` for instructions on editing rules at the command line.

Adjusting Spam Thresholds

"Thresholds" apply actions to messages based on their spam probability. By default, PureMessage is optimized to detect spam at a 50% threshold. Quarantining or marking messages below this threshold significantly increases the number of false positives. It is recommended that administrators begin tuning filters with a threshold of 50% to 60% to best ensure that legitimate messages reach recipients and that spam messages are quarantined.



Related concepts

[Lists](#) (page 252)

[Spam Detection](#) (page 271)

[Managing Anti-Spam Rules](#) (page 124)

Related information

[re.rules](#)

[pmx-spam](#)

Configuring the Policy Script

Messages processed by PureMessage are passed through the policy filter. The policy filter compares characteristics of the message against policy tests, and depending on the outcome of the test, applies the associated message-handling action. Together, tests and actions are referred to as "rules".

Policy actions are the components of rules that determine what happens to a message that passes the test. Multiple actions can be specified. General actions include:

- Accept the Message: Deliver the message to its envelope recipients.
- Copy Message to Quarantine: Copy the message to the quarantine; deliver the message to the intended recipient(s).
- Add Header and Deliver Message: Changes the message header; delivers the message to the intended recipient(s).

See “Policy Actions”, in the *Manager Reference*, or the `pmx-policy` man page, for a specific list of actions available for each module.

Related concepts

[About Actions](#) (page 93)

Related information

[pmx-policy](#)

1.2.3 Optimizing Performance

- Latest Version: Ensure that you are using the latest version of the PureMessage software by navigating to the **Support > Available Updates** page in the PureMessage Manager, selecting the repository from the **Package Repositories** drop-down list, and clicking **Query**. If any of the packages can be upgraded, you can do so by running `pmx-setup` at the command line.
- PureMessage Service Configuration: Be sure that the PureMessage services running have sufficient RAM and CPU available. For recommendations, see “Memory and CPU Requirements” in the Prerequisites section of the Installation Guide.
- Concurrent Interpreters: In the `pmx.conf` file, the `concurrency_limit` setting determines the maximum number of Perl interpreters that may be allocated to filter requests. When that value is reached, the `concurrency_limit_action` setting determines the action. Tuning these options can result in more efficient memory usage.
- DNSBL (DNS Black List) or DNS Lookups by the Spam Filter: If the spam filter is configured to perform [DNSBL](#) (DNS Black List) or DNS checks, PureMessage performance is strongly affected by the connection speed between PureMessage and the DNS server. For optimal performance, install a local caching DNS server. To disable network checks entirely, set the enabled option in `etc/spam.d/dnsbl.conf` to “no”. See “About Anti-Spam Rules” in the Policy Configuration section of the *Administrator’s Reference* for more information.
- IP Blocking: Ensure that you are using IP blocking, either at the MTA level or in the policy. The most effective way is to use IP blocking at the MTA level. This is set on the **Local Services** tab of the PureMessage Manager. Ensure that the **IP Blocker Service** is running, and then click **MTA IP Blocking** on the sidebar to ensure that this feature is enabled. Additionally, reverse DNS checks that detect dynamic IP addresses can be enabled. For more information, see “Enabling or Disabling MTA IP Blocking” in the *Manager Reference*.

A lesser, but still very significant impact on performance is to use IP blocking in the policy. This option also requires that the **IP Blocker service** is running. On the **Policy** tab of the Manager, add a main rule, setting the test as **Message is from blocked IP**, and ensuring that the action is **Stop processing**.

- Quarantine Database Options: PureMessage offers two database options: PostgreSQL and CDB (Common Database). PostgreSQL is installed by default, and it is required for multiple-server deployments requiring a centralized quarantine and PureMessage reporting. The CDB data store is a flat-file database that can be used in single-server implementations where reports are not a requirement. See “How do I change the quarantine indexing database?” in the Frequently Asked Questions for information about switching from the default database, PostgreSQL, to CDB.

- PostgreSQL and Kernel Tuning: It may be necessary to tune the PostgreSQL server kernel, depending on your message volumes and server configuration. See “Tuning PostgreSQL for PureMessage” in the Sophos Knowledgebase and PostgreSQL’s own documentation for more information.
- Other Recommendations:
 - If you originally installed PureMessage prior to version 5.2.1, ensure that you are using the End User Web Interface (EUWI) with the direct method by editing the `/opt/pmx6/etc/enduser/enduser_ui.conf` file as described at the end of either the Single-Server Upgrade or the Multi-Server Upgrade pages in the *Getting Started Guide*.
 - If possible, perform address verification at the MTA level (greatest performance improvement) or in the policy (lesser, but still improved performance).
 - If your organization’s policies allow it, discard high-probability spam, such as 90% and greater, or even just 99%, spam probability.
 - Ensure that you have implemented the settings for PostgreSQL suggested in the tuning guide that is included in the Sophos Knowledgebase.
 - If you are using sendmail and have a problem with load spikes, consider switching to Postfix.
 - Ensure large lists and maps are CDB-based.
 - Ensure that your logs are being rotated.

Related concepts

[Memory and CPU Requirements](#) (page 23)

[Changing the Quarantine indexing database](#) (page 283)

[About Anti-Spam Rules](#) (page 273)

Related information

[Tuning PostgreSQL for PureMessage](#)

[pmx.conf](#)

[DNS server software](#)

Configuring a Local Caching DNS Server

PureMessage performance is strongly correlated with the performance of the DNS server when the Anti-Spam filter is configured to perform DNS lookups (recommended). For optimal performance, install a local caching DNS server (see <http://www.dns.net/dnsrd/servers/> for more information).

By default, the Anti-Spam filter spreads DNS-based checks across all nameservers in `resolv.conf` regardless of their priority. To configure PureMessage to use only the local caching nameserver for these checks, add the following entry to `pmx/etc/spam.d/net.conf` within the `<plugin>` block and restart `pmx-milter`:

```
dns_servers = 127.0.0.1
```

This allows PureMessage to use the fastest available DNS server exclusively, while the system and MTA continue to use a prioritized list of nameservers for redundancy.

Related information

[DNS server software](#)

1.3 Installing PureMessage

This document describes how to install, configure and implement PureMessage. It is recommended that you first read the “Deployment Strategies” section of this guide, which provides examples of single-server and multi-server configurations, along with suggestions for phasing in PureMessage installations. For an overview of PureMessage and basic operating instructions, see the “Quick Reference Guide”.

Important

You must perform all installations as the root user.

Digitally Signed Installations

PureMessage uses certificate validation to verify the authenticity of installations, upgrades, and all subsequent software and data updates. Downloads from Sophos are signed with a private key that matches the public key embedded in the PureMessage validator application. This offers protection in the event that someone attempts to spoof the Sophos website to make it appear as though you are downloading from a trusted source. If a data or software update is interrupted, a partial update is not applied. This ensures the validity and completeness of the download.

In addition, you can configure PureMessage to notify you via email when a download fails the verification. This is done by enabling email notifications for scheduled jobs (see the `scheduler.conf` man page for more information).

Related concepts

[Deployment Strategies](#) (page 6)

[Quick Reference Guide](#) (page 65)

Related tasks

[Validating the Installation Script](#) (page 36)

Related information

[scheduler.conf](#)

1.3.1 Prerequisites

Before installing PureMessage, ensure that you meet the requirements described in the sections that follow. In addition to a supported operating system and a supported mail transfer agent (MTA), PureMessage has various other requirements, including certain standard system utilities, sufficient hard disk space, and access to a number of specific ports.

Supported Platforms

PureMessage's list of currently supported platform versions, see the Sophos “Software lifecycle” web page, <https://community.sophos.com/kb/en-us/119019>

Note

PureMessage may also be run on a virtual operating system using either VMware ESX (for Linux) or Solaris 10 Containers (for Solaris). The virtualization software must support your operating system and version.

- Linux [32-bit and 64-bit]
 - Ensure that you have the latest available updates for your operating system installed.
 - PureMessage is only supported on the original kernel provided by the vendor as part of the distribution.
 - If you are installing PureMessage on a 64-bit operating system, ensure that the 32-bit compatibility libraries (ia32-libs) are installed.
 - Ensure `libnss-mdns` is installed on your distribution.
 - The installer may warn that you are not on a supported distribution. If the above conditions are met you should be able to proceed with the installation.

See [Installing PureMessage on a 64-Bit Operating System](#) in the Sophos Knowledgebase for more information.

- Solaris
 - Open File Limitations: Ensure that the soft limit for the maximum number of system file descriptors in a single process is at least 1024 (`ulimit -n` reports the current setting). To change it, add the following two lines to your `/etc/system` file:

```
set rlim_fd_cur = 1024 set
rlim_fd_max = 4096
```

- On all versions, `libiconv` must be installed.

More detailed information on setting kernel parameters can be found in “Tuning PostgreSQL for PureMessage 5.x” in the Sophos Knowledgebase.

Important

PureMessage may not function correctly on supported operating systems if the locale is not set to English (`en_US.utf8`).

Related information

[Tuning PostgreSQL for PureMessage 5.x](#)

[Installing PureMessage on a 64-Bit Operating System](#)

[Red Hat glibc packages](#)

[Sophos Software Lifecycle](#)

Supported Mail Transfer Agents

Use PureMessage with one of the following mail transfer agents:

- Postfix: PureMessage includes a distribution of Postfix version 2.8.2, which is installed as part of a Full Installation. You can install this version from the PureMessage installer, or configure an existing Postfix installation to work with PureMessage.

- **Sendmail:** PureMessage also includes a distribution of sendmail version 8.14, which can be installed from the PureMessage installer. You can install PureMessage sendmail on the same server that you install PureMessage, or you can install it on a different server. Alternatively, you can configure PureMessage to work with an existing sendmail or Sendmail Switch installation. The following versions are supported:
 - Sendmail Switch v2.6 or later
 - sendmail v8.11.0 or laterSee 'When should I use an existing sendmail, Postfix or Java System Messaging Server (JSMS) installation?' in the PureMessage FAQ for more information.
- **Oracle Communications Messaging Exchange Server:** Although a version of Oracle Communications Messaging Exchange Server is not bundled with the product, PureMessage supports this MTA (if patched to include milter support). In order to use this MTA with PureMessage, you must configure it to communicate directly with the PureMessage milter. For instructions, see “Configuring PureMessage for Oracle Communications Messaging Exchange Server” in the Sophos Knowledgebase.

Important

Do not assign a Mail Transfer Agent role during installation.

Related information

[Which Mail Transfer Agent Should I Use?](#)

[Sophos software lifecycle page](#)

[Configuring PureMessage for Oracle Communications Messaging Exchange Server](#)

Other Prerequisites

Before installing PureMessage, ensure that you meet the following assorted requirements:

PostgreSQL

If you are using the PostgreSQL database included with PureMessage to index quarantine metadata, you may need to manually tune your system after installation. See “Tuning PostgreSQL for PureMessage” in the Sophos Knowledgebase for more information.

For PureMessage deployments with high mail volume or a large number of end users, installing this component on a separate server with a fast SCSI disk array (other than RAID 5) is recommended.

Disk Space

The PureMessage installation requires approximately 500 MB of disk space. In addition, the quarantine and log files created during PureMessage operation require an indeterminate amount of disk space that depends on the volume of mail that is processed and/or quarantined. The log and quarantine directories, however, may be stored on a different partition from the one where PureMessage is installed. PureMessage does not support the sharing of files between servers via Network File System (NFS).

A Local Mailer

PureMessage requires a local mailer program (for example, Procmailer (Linux) or `/bin/mail` (Solaris)). Local mailers (programs used by mail transfer agents to handle local mail delivery) are included with most UNIX distributions. Sendmail comes with its own local mailer, "mail.local", which can be used if the operating system does not provide one.

A Supported Browser for the PureMessage Manager

The PureMessage Manager web interface, the End User Web Interface (EUWI) and the Groups Web Interface support recent versions of Internet Explorer, Firefox, and Chrome.

A umask Setting of 0022 for the Root User

PureMessage assumes a umask setting of 0022 for the root user. If the umask on a PureMessage server is set to something more restrictive (for example, 0077), then installations and upgrades could fail. If your organization requires a umask setting other than 0022, you must switch to 0022 for the purpose of PureMessage installation and during all future upgrades.

Related concepts

[Memory and CPU Requirements](#) (page 23)

Related information

[Tuning PostgreSQL for PureMessage](#)

Utilities

The PureMessage installation relies on a number of standard system utilities. These are included with most Unix distributions. To check if these utilities are installed, enter `which utility_name` at the command line. Required utilities include:

- **awk**: This pattern-matching program is required if you want to use the log search functionality that is available through PureMessage's Groups Web Interface. (See also, `mkfifo`.)
- **cs**: Perl must have access to the C shell binary in order to expand file glob patterns. (The PureMessage user's default shell, however, must be a Bourne-compatible shell such as `bash`.) Typically, `cs` is available by default on all supported platforms. However, on some systems, installation of `cs` is optional, and therefore it may not be present. A suitable replacement for `cs` is `tcsh`, which can be obtained in source form from <http://www.tcsh.org/Home>. Note that `cs` must be symlinked to `tcsh` if you install the latter in lieu of the former.
- **m4**: The m4 macro processor is required for rebuilding sendmail and Postfix configuration files and is used in the mailer's startup script. It is not used by other parts of PureMessage. If your platform does not provide `m4` by default, you can build GNU m4 from source. GNU m4 is available from <http://directory.fsf.org/GNU/gnum4.html>.
- **make**: Most supported systems provide a `make` utility by default. If your system does not have it, you can build GNU `make` from source. GNU `make` is available at <http://directory.fsf.org/GNU/make.html>.
- **mkfifo**: The UNIX program used to create named pipe special files. This utility is a prerequisite for using the the log search functionality that is available through PureMessage's Groups Web Interface. (See also, `awk`.)

- **random, urandom:** PureMessage requires `/dev/random` and `/dev/urandom` devices. On Solaris, these devices may not be present. To add them, see "27606: Differing `/dev/random` support requirements within Solaris [TM] Operating Environments" at <http://sunsolve.sun.com>.
- **useradd, userdel, groupadd, groupdel:** `useradd` and related programs are available by default on many supported systems. If you are installing PureMessage as an NIS-based user, ensure that the corresponding user and group manipulation utilities are available in your `PATH`.
- **uudecode, or perl:** `uudecode` is normally available on all supported platforms. This program is used by the initial installer script for "bootstrapping". If `uudecode` is not found, Perl (version 4 or later) will be used if available. If neither is available, you can install the GNU `sharutils` package, which is available from: <http://directory.fsf.org/sharutils.html>.

Related information

[tssh](#)

[GNU M4 - Macro processor](#)

[make](#)

[sharutils](#)

Port Usage

PureMessage contains a number of discrete components (for example, the `mlt` interface, the Manager, the End User Web Interface, the PostgreSQL database), and interacts with external components (such as the Postfix or `sendmail` mail transfer agents). These components can all be deployed on the same server, or can be distributed among multiple servers. In either case, the components must have the ability to communicate with one another on various TCP ports.

Postfix	By default, Postfix receives incoming SMTP connections on port 25 and communicates with PureMessage using the <code>content_filter</code> mechanism on ports 10025 and 10026.
sendmail	By default, <code>sendmail</code> receives incoming SMTP connections on port 25 and communicates with PureMessage using the <code>mlt</code> protocol on TCP port 3366.
Oracle Communications Messaging Exchange Server	By default, Oracle Communications Messaging Exchange Server receives incoming SMTP connections on port 25 and communicates with PureMessage through port 3366. For more information, see "Configuring PureMessage for Oracle Communications Messaging Exchange Server" in the Sophos Knowledgebase.
PureMessage Manager	Day-to-day management of the PureMessage system using the Manager interface occurs using a TCP connection to port 18080 (configurable) on the PureMessage server. This port must be open between the administrator's workstation and the PureMessage server.
SSH or Telnet	Many administrators choose to work with PureMessage from the command line, which requires <code>ssh</code> (port 22) or <code>telnet</code> (port 23) access to the PureMessage server.
DNS and DNSBL checks	PureMessage can be configured to perform a variety of network checks as part of its spam

	<i>heuristic</i> analysis. These include DNS, DNSBL (DNS Black List), and SXL lookups (port 53 UDP and TCP).
Updates	PureMessage updates occur via HTTP over port 80. (Updates can also be installed from a local tarball, if required.)
Central Server Management	The Central Server Management (Server Groups) aspect of PureMessage identifies PureMessage servers on the network via UDP queries on port 18080. Synchronization between the PureMessage servers occurs over port 18080. Quarantine consolidation occurs using scp, which runs over port 22.
End User Web Interface	By default, the End User Web Interface runs on port 28443. Users must be able to connect to this port on the server from their workstations.
Groups Web Interface	By default, the Groups Web Interface runs on port 28443. Group administrators must be able to connect to this port on the server from their workstations.
PostgreSQL	By default, PostgreSQL listens on port 5432. If PureMessage is running on a separate server, it must access the PostgreSQL host on this port.

Memory and CPU Requirements

The following recommendations assume a mail server that is processing a volume of approximately 120,000 messages per hour.

For handling extremely large volumes of mail, you can get very good scalability by increasing the number of central processing units (CPUs) on the PureMessage server and by running different filters on separate PureMessage servers.

Minimum Configuration

This configuration is suitable for testing or for sites with very low email volume.

- CPU (Solaris): Sparc 2 x 1GHz or higher UltraSPARC III or equivalent
- CPU (Linux): 1 x 2GHz Intel Pentium 4 or equivalent
- Memory: 2 GB RAM
- Network: 100 Mb
- DNS: A server on the same network as PureMessage and at least one backup server with identical configuration (to avoid delays in mail delivery due to administrative stoppages).
- OS: A supported operating system as described in the “Supported Platforms” section.

Recommended Configuration

- CPU: 2 x 3GHz Intel Xeon or better
- Memory: 4 GB RAM with Postfix; 4 GB RAM with Sendmail
- Network: 100 Mb or better
- Disk: Ultra Wide SCSI/Ultra ATA100 RAID0 or better
- DNS: Two servers on the same subnet as PureMessage capable of 10-50ms lookups with equal MX priority (plus two or more identically configured backup servers).

- OS: In addition to the minimal configuration described above, full SMP support in OS; scalable pthreads implementation built on kernel threads or light-weight processes.

The per-process defaults or kernel parameters should be tuned so that the pmx process is able to access at least the following resources:

- 1024 file descriptors (or $\geq 10 \times \text{concurrency_limit}$)
- 1024 open files (or $\geq 10 \times \text{concurrency_limit}$)
- 256 threads per process (or $\geq 1.5 \times \text{concurrency_limit}$)
- 16 MB maximum process stack segment size
- unlimited maximum process data segment size
- unlimited memory addressable by process

Many of the settings described above can be adjusted using the `ulimit` command. Some will need adjustment of kernel parameters, and a recompiled kernel on some systems.

Related concepts

[Supported Platforms](#) (page 18)

1.3.2 Obtaining the PureMessage Distribution

PureMessage is distributed via three methods: the PureMessage download server, CD-ROM, and tarball. A general description of these media follows. Proceed to “PureMessage Installation” or “Upgrading to PureMessage” for installation or upgrade instructions.

Related concepts

[PureMessage Installation](#) (page 28)

[Upgrading PureMessage](#) (page 52)

PureMessage Download Server

If installing PureMessage from a download server, you must allow HTTP access for downloads over port 80 to the host where the PureMessage installation is being performed.

If you use a proxy server to access the internet, ensure that you have set the `HTTP_proxy` environment variable. `HTTP_proxy` should be set to a value of the form `http://proxyhost.example.com:PORT`, or `http://username:password@proxyhost.example.com:PORT` for proxy servers that require a username and password.

CD-ROM Distribution

The PureMessage distribution is also provided on CD-ROM. There is a directory for each platform beneath the root of the disc. Each platform-specific directory contains a full PureMessage distribution. The root directory of the disc contains text versions of the documentation files that are most commonly used during installation. The complete HTML version of the documentation is stored in the `html` directory.

Note

When installing from a CD-ROM that does not contain the latest version of PureMessage, it is recommended that you update your packages immediately following installation to ensure that you are using the most recent anti-spam and anti-virus definitions. For more information, see the “Available Updates” instructions in the Support Tab section of the *PureMessage Manager Reference*.

Related concepts

[Support Tab](#) (page 189)

Tarball Distribution

PureMessage tarballs are located in the `tarball` directory on the PureMessage download server (`pmx.sophos.com`). Use a file transfer utility, such as `wget`, to download the tarball. Log in as the PureMessage user, and extract the tarball into a temporary directory before proceeding.

Downloading and Extracting the Tarball

1. Log in as the PureMessage user ('pmx', by default).
2. Change to your system's temporary directory:

```
cd /tmp
```

Note

You must extract the tarball into a directory where the PureMessage user has read and execute permissions (such as `/tmp`), or explicitly grant permissions after extracting the tarball.

3. From the Sophos website, retrieve the PureMessage version that matches your platform:

```
wget pmx-static2.sophos.com/tarball/pmxrepo-version-platform.tar
```

4. Ensure that the ownership of this file is set to `pmx` for the both the user and the group.
5. Extract the contents of the tarball:

```
tar -xvf pmxrepo-version-platform.tar
```

6. If you are installing PureMessage, copy the installation script that Sophos sent to you via email to the same directory where you extracted the tarball (for example, `/tmp`).

Installing or Upgrading PureMessage from a Tarball

Once the tarball has been extracted according to the instructions shown above, choose the instructions that match your upgrade scenario:

- “PureMessage Installation” (if you are installing PureMessage from a tarball).
- “Upgrading to PureMessage 5.6.1 from a Tarball Distribution.”
- The “PureMessage 6 Installation” instructions in “Upgrading to PureMessage 6 from a Tarball Distribution” (if you are upgrading from 5.6.1 to the latest version of PureMessage).

Important

Following an installation or upgrade, you must reconfigure PureMessage so that it retrieves future updates and upgrades from the correct Sophos repository.

1. Once the tarball installation or upgrade has completed, return to the main menu of the installer.
2. Select **Upgrade PureMessage Components**, then select **Change Repository**.
3. In the **Change Repository** text box, enter:

Version 5.x :

```
http://pmx-dynamic.sophos.com/pmx/mainline/<Platform>/
```

Version 6 :

```
http://pmx-dynamic.sophos.com/pmx/v6/mainline/<Platform>/
```

4. Select **OK**, and press **Enter** .
5. Select **Return to the main menu**, and then **Exit the installer**.

Related concepts

[PureMessage Installation](#) (page 28)

[PureMessage 6 Installation](#) (page 57)

[Upgrading PureMessage](#) (page 52)

[Upgrading to PureMessage 5.6.1 from a Tarball Distribution](#)

Perform the following steps to complete a tarball upgrade on a single server configuration. If you are upgrading multiple PureMessage servers, also refer to “Multi-Server Upgrade” in “Upgrading to PureMessage 5.6.1”

Note

- You must perform this procedure as the root user.
- This procedure assumes that you have completed the “Downloading and Extracting the Tarball” steps in “Tarball Distribution.”

1. Run the PureMessage upgrade command. Assuming, the default installation directory (`/opt/pmx`) is used, and the tarball was extracted to `/tmp/pmxrepo`, run :

```
/opt/pmx/bin/pmx-setup --repo=file:///tmp/pmxrepo
```

2. On the main menu, select the **Upgrade PureMessage Components** option, and then press **Enter** .
3. Select **Upgrade components**, and then press **Enter** .
4. You are prompted to stop PureMessage before continuing the installation. Select **Yes**, and press **Enter** .

Updating the PureMessage components may take a few minutes. When all packages are upgraded successfully, press **Enter** .

5. Once the tarball upgrade has completed, select **Change Repository**.

6. In the **Change Repository** text box, enter:

```
http://pmx-dynamic.sophos.com/pmx/mainline/<Platform>/
```

Select **OK**, and press **Enter**.

7. Select **Return to the main menu**, and then press **Enter**.
8. Select **Exit the installer**, and then press **Enter**.

Related concepts

[Upgrading PureMessage](#) (page 52)

Related tasks

[Validating the PureMessage Upgrade Program](#) (page 64)

[Multi-Server Upgrade](#) (page 64)

Upgrading to PureMessage 6 from a Tarball Distribution

Perform the following steps to complete a tarball upgrade on a single server configuration. This is not a typical upgrade. You will install PureMessage version 6 alongside your existing 5.6.1 installation, and, optionally, migrate data from version 5.6.1 to the newly installed version 6.

If you plan to migrate data from your existing installation to version 6 using the PureMessage migration script, you must already be upgraded to 5.6.1. For more information, See “Upgrading to PureMessage 5.6.1 from a Tarball Distribution” and “Migrating from Version 5.6.1 to 6.”

If you are upgrading multiple PureMessage servers, also refer to “Multi-Server Upgrade” in “Upgrading to PureMessage 6.”

Important

- You must perform this procedure as the root user.
- This procedure assumes that you have completed the “Downloading and Extracting the Tarball” steps in “Tarball Distribution.”

1. Ensure that PureMessage 5.6.1 is stopped.
2. Obtain the PureMessage installation script for version 6 from your Sophos representative, and copy it to the same directory where you extracted the tarball.
3. Run the version 6 installation script:

```
sh puremessage-<VersionNumber>-<Platform>.sh
```

4. Complete the installation of PureMessage 6. For instructions, see [PureMessage Installation](#). When you are finished, perform the remaining steps (below).

Important

Following an installation or upgrade, you must reconfigure PureMessage so that it retrieves future updates and upgrades from the correct Sophos repository.

1. Once the tarball upgrade has completed, return to the main menu of the installer.
2. Select **Upgrade PureMessage Components**, then select **Change Repository**.
3. In the **Change Repository** text box, enter:

```
http://pmx-dynamic.sophos.com/pmx/v6/mainline/<Platform>/
```

4. Select **OK**, and press **Enter**.
5. Select **Return to the main menu**, and then **Exit the installer**.

During installation, a symlink called “pmxrepo” is created in `/opt/pmx/home/` that points to `/tmp/pmxrepo`. Once your tarball installation is complete, you must remove this symlink to ensure that updates are retrieved from the repository that you specified in the procedure above.

Related concepts

[PureMessage Installation](#) (page 28)

[Upgrading PureMessage](#) (page 52)

Related tasks

[Upgrading to PureMessage 5.6.1 from a Tarball Distribution](#) (page 26)

[Migrating from Version 5.6.1 to 6](#) (page 60)

[Validating the PureMessage Upgrade Program](#) (page 64)

[Multi-Server Upgrade](#) (page 64)

1.3.3 PureMessage Installation

Use the PureMessage installer to install and deploy PureMessage in varying configurations. The **Full Installation** option installs PureMessage on a single server. The **Custom Installation** option provides the flexibility to assign server roles to any number of servers in a network.

Before running the installer, review the PureMessage “Prerequisites” section.

Note

While many terminal applications and environment variables will work with the PureMessage installer, Sophos has tested and supports using xterm, putty and the console with a `$PMX_TERM` environment variable of either `xterm` or `vt100`. If you have trouble with your installer’s display, use one of these combinations.

PureMessage includes a menu-based installer that streamlines the installation process. This installer launches in a console window. The window is partitioned into two sections; the left side displays installation options, and the right side displays help text. Use the following keyboard commands to navigate the installer:

- **Tab** : moves cursor from one option to another
- **Left/Right Arrow** : shifts focus between the left and right panes of the installer
- **Up/Down Arrow** : scrolls the text in the right pane of the installer (if the right pane has focus)

- **Space Bar** : selects the highlighted option or check box(es)
- **Enter** : accepts the selected options or check box(es)

Important

Sophos strongly recommends that you do not resize the installer console window once the PureMessage installer launches. Resizing the console window may cause the installer to terminate. Before installing, stop any process that uses port 25, including existing versions of Postfix or sendmail. Regardless of whether you configure PureMessage to work with an existing mail transfer agent (MTA) or the sendmail or Postfix version that is bundled with PureMessage, you must start the designated MTA after installation is complete.

Related concepts

[Prerequisites](#) (page 18)

Full Installation

The **Full Install** option installs a complete PureMessage system on a single server.

To install PureMessage on two or more servers, see the “Custom Installation” section.

A full installation consists of the following:

- Centralized Server Manager (CSM)
- Database server
- End User Web Interface/Groups Web Interface
- Mail filtering
- PureMessage Postfix
- Sender History Database

Follow the steps below to complete a **Full Install**:

1. Obtain the Distribution. See the “Obtaining the PureMessage Distribution” section of the Installation Guide for more information.
2. Run the installer. On the command line, enter:

```
sh puremessage-<VersionNumber>-<Platform>.sh
```

For the command above and subsequent installation commands, enter the relevant *VersionNumber* and *Platform*.

Note

PureMessage uses certificate validation to verify the authenticity of software and data updates. If you want to exercise the utmost caution, you can verify the installation script itself. For more information, see “Validating the Installation Script”.

3. Welcome to PureMessage. Proceed with installation? Press **y** (for yes) to proceed with the PureMessage installation, and then press **Enter**.
4. Enter a directory for the PureMessage installation. Select the directory where PureMessage and its supporting files will be installed. The default location is `/opt/pmx6/`. The directory must have at least 500 MB of free space. Additional space is required for log files and messages stored in

the PureMessage quarantine. Enter a location at the prompt and press **Enter** , or press **Enter** to select the default installation location.

Important

It is not recommended that you move any of the directories that are located directly beneath `/opt/pmx6/` in the PureMessage installation directory. Directories such as `postfix`, `postgres`, etc contain binary files and libraries that are required for PureMessage updates. Moving or symlinking these directories could cause upgrades to fail.

5. Accept the Sophos End User License Agreement. Use the **Page Up** and **Page Down** keys to view the entire user agreement. Be sure not to resize the console window. Press **Tab** to move the cursor to the **I accept the licensing terms** box. Press **Space Bar** to select this option. Press **Tab** to select **OK**, and then press **Enter** .
6. Select the Full Install option. Press **Tab** to select **Full Install**, and then press **Enter** . The installer will guide you through the remainder of the installation process. After completing each step, use the **Tab** key to highlight **<Next Question>**, and press **Enter** . At any point in the process, you can return to the previous question or go back to the main installation menu. Refer to the documentation in the right pane of the installer for information about the currently displayed installation option.
7. Complete the installation. Once you have finished entering the installation information, you are prompted to **Proceed with the installation**. Press **Enter** .
8. Exit the PureMessage Installer. Once you have completed the steps, exit the installer.
9. Start PureMessage. At the command line, run:

```
pmx start
```

10. Start the mail transfer agent. Once you have completed the installation and started PureMessage, you must start the mail transfer agent (MTA) that will be used with PureMessage. You can start the MTA in the PureMessage Manager or from the command line.
 - **Manager:** On the **Local Services** tab, under **Background Services**, select the check box next to the SMTP service and click **Start**.
 - **Command Line:** As the 'pmx6' user, run `pmx-service start smtp`.

If you want to use PureMessage with an existing mail transfer agent, see “Configuring External Mail Transfer Agents” in the Post-Installation Procedures section of the *Getting Started Guide*.

Note

During installation, PureMessage automatically adjusts the PostgreSQL database in an attempt to improve performance. If the adjustments are successful, PureMessage displays a confirmation message. Depending on the shared memory configuration of your system, however, additional manual steps may be required.

If your system settings are not within the recommended range, PureMessage tries to make the necessary adjustments. In some cases, the updated settings are saved to file named `postgresql.conf.recommended`. You can accept these settings by renaming the file to `postgresql.conf`. If your system does not meet even the minimum requirement for shared memory, a error message is displayed. For more information, see “Tuning PostgreSQL for PureMessage” in the Sophos Knowledgebase.

Related concepts

[Custom Installation](#) (page 31)

[Obtaining the PureMessage Distribution](#) (page 24)

[Configuring External Mail Transfer Agents](#) (page 45)

Related tasks

[Validating the Installation Script](#) (page 36)

Related information

[Tuning PostgreSQL for PureMessage](#)

Custom Installation

Use the **Custom Installation** option to install PureMessage on two or more servers. This option provides the flexibility to assign server [roles](#) (described below) to any number of servers in a network.

Server roles can be configured in a variety of combinations. You can distribute these roles among multiple servers however you choose. For example, in a two-server configuration, you may want to install the Centralized Server Manager and Database Server roles on the first server and the Mail Filter and Mail Transfer Agent roles on the second server. See the “Deployment Strategies” section of the *Getting Started Guide* for other configuration scenarios.

Select from the following roles:

Centralized Server Manager (CSM)

The Centralized Server Manager role installs the main interface for configuring the PureMessage network. The server running the CSM role must be the first server that is set up. This role *must* be installed on the same server as the Database Server role.

Database Server

The Database Server role installs the PureMessage database for use with DBMS-based quarantines and for PureMessage reporting functions. The Database Server role and the Centralized Server Manager role *must* be installed on the same server. Sophos does not recommend installing both the Database Server role and the Mail Filter role on the same server. In multi-server deployments, you must set up the Centralized Server Manager role and the Database Server role before installing any Mail Filter role to ensure proper synchronization of PureMessage resources.

End User Web Interface Server/Groups Web Interface Server

The End User Web Interface makes it possible for end users to manage their own quarantine, whitelists, and blacklists. Smaller PureMessage installations may run this role with the Database Server role, while larger installations should distribute these roles across multiple machines. The Groups Web Interface role, which is installed along with the End User Web Interface role, gives a global administrator the option of delegating administrative responsibilities to “group” administrators based on groups/domains and/or roles.

Mail Filter

The Mail Filter role installs the PureMessage [militer](#). This role is typically used in combination with the Mail Transfer Agent role to scan and filter mail for viruses and spam. Larger installations will likely have several mail filter servers.

Mail Transfer Agent

The Mail Transfer Agent server role installs the third-party mail transfer agent (MTA) software (sendmail or Postfix). The MTA receives incoming SMTP connections and sends the messages through the PureMessage mail filter. The Mail Transfer Agent role is typically installed on the same server as the Mail Filter role. In a multi-server deployment, the MTA role must also exist on the server where the CSM role is installed.

During a Custom Installation, you are prompted to choose either sendmail or Postfix as the mail transfer agent. The PureMessage installer only installs the versions of PureMessage and sendmail included with the PureMessage distribution. See “Configuring an External Sendmail Installation” or “Configuring an External Postfix Installation” in the Getting Started guide for instructions on configuring an existing sendmail, Sendmail Switch or Postfix installation. If you are not sure which type of installation to use, see “Which Mail Transfer Agent Should I Use?” in the PureMessage Sophos Knowledgebase.

The Oracle Communications Messaging Exchange Server mail transfer agent is not part of the PureMessage distribution and must be installed separately. For information about configuring this MTA, see “Configuring PureMessage for Oracle Communications Messaging Exchange Server” in the Sophos Knowledgebase.

Note

During installation, PureMessage automatically adjusts the PostgreSQL database in an attempt to improve performance. If the adjustments are successful, PureMessage displays a confirmation message. Depending on the shared memory configuration of your system, however, additional manual steps may be required.

If your system settings are not within the recommended range, PureMessage tries to make the necessary adjustments. In some cases, the updated settings are saved to file named `postgresql.conf.recommended`. You can accept these settings by renaming the file to `postgresql.conf`. If your system does not meet even the minimum requirement for shared memory, a error message is displayed. For more information, see “Tuning PostgreSQL for PureMessage” in the Sophos Knowledgebase.

Sender History Database

PureMessage creates and maintains a Sender history database in order to protect you from snow shoe spam attack. You must install this role on a separate system, although it is possible to club this role with existing PureMessage roles. Sophos does not recommend to club this role with Database Server role as it may degrade the performance.

To install Sender History Database role and configuration of Delay Queue, click [Installing Sender History Database and Configuring Delay Queue](#).

Related concepts

[Custom Installation](#) (page 31)

[Deployment Strategies](#) (page 6)

[Configuring an External Sendmail Installation](#) (page 45)

[Configuring an External Postfix Installation](#) (page 49)

Related information

[Which Mail Transfer Agent should I Use?](#)

[Tuning PostgreSQL for PureMessage](#)

Configuring PureMessage for Oracle Communications Messaging Exchange Server

Custom Installation Example: Two Servers

The following describes one possibility for configuring PureMessage using the Custom Install option. See “Server Deployment Options” for other examples.

In this installation, the first server hosts the Centralized Server Manager role, the Database Server role and the End User Web Interface Server role. The second server hosts both the Mail Transfer Agent role and the Mail Filter role.

Important

- It is recommended that you install the Mail Transfer Agent role on both the central server and the edge server that you plan to use as your mail server (as described in this example). If you intend to install the Mail Transfer Agent role on a server that is separate from the one on which the Centralized Server Manager role is installed, you must specify the hostname and port number of the mail server so that the central server will be able to send it administrative notifications and messages that are released from the quarantine. You will be prompted to provide the hostname and port as one of the Custom Installation steps.
- If you have the Mail Transfer Agent role (Postfix) installed on the same server as the Centralized Server Manager role, subsequent MTA role installation on different servers will inherit Mail Transfer Agent (Postfix) configuration from that Centralized Server Manager when you connect with it during the installation process. The installation wizard will not prompt you for Postfix configuration. If required, you can change your Postfix configuration by editing related configuration files (main.cf) after installation.
- To ensure consistent reporting across multiple servers, use ntp synchronization with ntpd so that the system clock is the same on all servers. In addition, all servers must be set to the same time zone for accurate reporting.
- It is recommended that you specify the same username for the PureMessage user ('pmx6' by default) on each server in your deployment. If you use different usernames, you must manually configure the DSN string.

To install this multi-server configuration, complete the steps in the order described below.

- To configure the first server:
 1. Obtain the Distribution. See the “Obtaining the PureMessage Distribution” section of the Installation Guide for more information.
 2. Run the PureMessage installer. On the command line, enter:

```
sh puremessage-<VersionNumber>-<Platform>.sh
```

For the command above and all subsequent installation commands, enter the relevant *VersionNumber* and *Platform*.

Note

PureMessage uses certificate validation to verify the authenticity of software and data updates. If you want to exercise the utmost caution, you can verify the installation script itself. For more information, see “Validating the PureMessage Installer”.

3. Welcome to PureMessage. Proceed with installation? Press **y** (for yes) to proceed with the PureMessage installation, and then press **Enter**.
4. Enter a directory for the PureMessage installation. Select the directory where PureMessage and its supporting files will be installed. The directory must have at least 500 MB of free space. Additional space is required for log files and messages stored in the PureMessage quarantine.

Enter a location at the prompt and press **Enter**. The default location is `/opt/pmx6/`. Press **Enter** to select the default installation location.
5. Accept the Sophos End User License Agreement. Use the **Page Up** and **Page Down** keys to view the entire user agreement. Be sure not to resize the console window. Press **Tab** to move the cursor to the **I accept the licensing terms** box. Press **y** (for yes) to accept the licensing terms. Press **Tab** to select **OK**, and then press **Enter**.
6. Select the custom installation option. Select **Custom Install**, and then press **Enter**.
7. Select roles for the first server. Select the following roles for the first server:
 - Centralized Server Manager
 - Database Server
 - End User Web Interface/Groups Web Interface Server
 - Mail Transfer Agent

Use the up and down arrow keys to highlight these server roles. Press **y** (for yes) to select each role. Press **Tab** to highlight **Install these roles**, and then press **Enter**.
8. Answer the series of questions. Follow the instructions in the console window. The questions asked are dependent on the server roles selected. The installer will guide you through the remainder of the installation process. After completing each step, use the **Tab** key to highlight **<Next Question>**, and press **Enter**. At any point in the process, you can return to the previous question or go back to the main installation menu. Refer to the documentation in the right pane of the installer for information about the currently displayed installation option.
9. Complete the installation. Once you have finished entering the installation information, you are prompted to **Proceed with the installation**. Press **Enter**.
10. Start PureMessage. Run the `pmx start` command.
- To configure the second server:
 1. Obtain the Distribution. See the “Obtaining the PureMessage Distribution” section of the Installation Guide for more information.
 2. Ensure that the CSM and PostgreSQL servers are running. Both the Centralized Server Manager role and the Database Server role must be running on the first machine for the second server installation to succeed.
 3. Run the installer. On the command line enter:

```
sh puremessage-<VersionNumber>-<Platform>.sh
```

4. Welcome to PureMessage. Proceed with installation? Press **y** (for yes) to proceed with the PureMessage installation, and then press **Enter**.
5. Enter a directory for the PureMessage installation. Select the directory where PureMessage and its supporting files will be installed. The directory must have at least

500 MB of free space. Additional space is required for log files and messages stored in the PureMessage quarantine.

Enter a location at the prompt and press **Enter**. The default location is `/opt/pmx6/`. Press **Enter** to select the default installation location.

Important

It is not recommended that you move any of the directories that are located directly beneath `/opt/pmx6/` in the PureMessage installation directory. Directories such as `postfix`, `postgres`, etc contain binary files and libraries that are required for PureMessage updates. Moving or symlinking these directories could cause upgrades to fail.

6. Accept the Sophos End User License Agreement. Use the up and down arrow keys to view the entire user agreement. Be sure not to resize the console window. Press **Tab** to move the cursor to the **I accept the licensing terms** box. Press **y** (for yes) to accept the licensing terms. Press **Tab** to select **OK**, and then press **Enter**.
7. Select the Custom Install installation option. Press **Tab** to select **Custom Install**, and then press **Enter**.
8. Select roles for the second server. Select the following roles for the second server:
 - Mail Transfer Agent
 - Mail Filter

Use the up and down arrow keys to highlight these server roles. Press **y** (for yes) to select each role. Press **Tab** to highlight **Install these roles**, and then press **Enter**.
9. Answer the series of questions. Follow the instructions in the console window. The questions asked are dependent on the server roles selected.
10. Complete the installation. Once you have finished entering the installation information, you are prompted to **Proceed with the installation**. Press **Enter**.
11. Exit the PureMessage Installer. Once you have completed the steps, exit the installer.
- Start the mail transfer agent. Once you have completed the installation and exited the installer, you must start the mail transfer agent (MTA) that will be used with PureMessage.

If you selected Postfix or sendmail during installation, you can start the MTA in the PureMessage Manager or from the command line.

- Manager: On the **Local Services** tab, under **Background Services**, select the check box next to the SMTP service and click Start.
- Command Line: As the 'pmx6' user, run `pmx-service start smtp`.

If you want to use PureMessage with an existing mail transfer agent, see “Configuring External Mail Transfer Agents” in the Post-Installation Procedures section of the *Getting Started Guide*.

Related concepts

- [Server Deployment Options](#) (page 8)
- [Obtaining the PureMessage Distribution](#) (page 24)
- [Configuring External Mail Transfer Agents](#) (page 45)

Related tasks

- [Validating the Installation Script](#) (page 36)

Validating the Installation Script

Once you have installed PureMessage, automatic certificate validation ensures the authenticity of subsequent software and data updates. As an added precaution, before installing PureMessage, you can verify the authenticity of the installation script itself.

To validate the installer:

1. Run the installer. At the command-line, enter:

```
sh puremessage-<VersionNumber>-<Platform>.sh
```

The PureMessage installer program (`pmx-setup`) is retrieved from the Sophos website.

2. Extract the validator. If PureMessage is installed in the default location (`/opt/pmx`) At the command line, enter:

```
/opt/pmx/bin/pmx-setup --check-validator
```

The installer acknowledges that you have chosen to verify the authenticity of the validation tool itself before proceeding with the installation. The validation file is extracted.

3. Obtain the `shasum` of the validation file. The `shasum` utility is used to calculate and verify SHA1 hashes. If you don't have `sha1sum`, you must obtain the appropriate version for your operating system. See the `shasum` documentation for specific command syntax.
4. Contact Sophos Technical Support and have a representative confirm that the checksum you have obtained is correct. See “Contacting Sophos” for more information.
5. Verify that your copy of `pmx-setup` is valid. At the command line, run the `pmx-validator` command as shown below:

```
<PathToValidator>/pmx-validator <PathToRepository> <PathTo_pmx-setup>
```

The exact file locations for this command are the `pmx-validator`, `Repository`, and `pmx-setup` entries displayed in the installer.

6. Continue with the installation. Once verification is complete, enter:

```
/root/pmx-setup
```

Related concepts

[Contacting Sophos](#) (page 74)

1.3.4 Post-Installation Procedures

This section discusses configuration options that are generally set immediately following an installation or upgrade. It also includes instructions on upgrading individual components, installing new components, creating configuration snapshots and uninstalling PureMessage.

Note

To ensure that your installation is functioning properly, see the “Testing PureMessage Operations” section of the Installation Guide.

Related tasks

[Testing PureMessage Operations](#) (page 41)

Setting Preferences for Unscannable Attachments

After completing an installation, as part of the configuration and tuning, be sure to set preferences for unscannable attachments. This can present a problem for administrators of new installations if the anti-virus behavior is contrary to their expectations. See “Unscannable Attachments” in the Policy Configuration section of the *Administrator's Reference* for more information.

Related concepts

[Unscannable Attachments](#) (page 279)

Configuring End User Web Interface Authentication

If you selected the **End User Web Interface/Groups Web Interface** role during PureMessage installation, you may want to reset the method for authenticating EUWI access. The following methods are available:

Email Session Authentication (Default)

This is the default authentication method. End users are prompted for their username and email address on their first visit. An automatically generated session ID key is emailed to the user as his or her password.

Flat-File Authentication

This optional method uses a flat-file database (a plain text file in a specific format, which can be encrypted) to store usernames and passwords. This method requires that the PureMessage administrator add all the usernames and their initial passwords to the sample end user password file, and then email the end users to notify them of their passwords.

LDAP Authentication

This optional method allows you to use an existing LDAP directory, such as Active Directory, Sun ONE Directory Server 5.2, or OpenLDAP, as the source for end user authentication.

Configuring Http/Https Access to the End User Web Interface

By default, end users connect to the End User Web Interface (EUWI) over an https connection on port 28443. Optionally, you can configure PureMessage to permit unsecured EUWI access over http on port 28080 as well.

To allow EUWI access via both http and https:

- At the command line, as the 'pmx6' user, run the following command:

```
ln -sf /opt/pmx6/etc/manager/httpd2/ssl/default.conf /opt/pmx6/etc/manager/httpd2/ssl.conf
```

- To reset to the default, https only, run the following command:

```
ln -sf /opt/pmx6/etc/manager/httpd2/ssl/http.conf /opt/pmx6/etc/manager/httpd2/ssl.conf
```


Note

The setting that you specify will also be used to access the Groups Web Interface on the same server.

Configuring Log Searching

If you have installed the latest version of PureMessage with the Postfix mail transfer agent (MTA), all of the basic requirements for log searches are enabled by default. Log searches are not supported for the sendmail or Oracle Communications Messaging Exchange Server MTAs. If you are using Postfix and have moved to the latest version by way of an upgrade, you can enable message log searching as follows:

1. If you want the “Subject” of a message included in search results, modify the PureMessage policy to include the following as the first line of Sieve code:

```
pmx_mark "s" "%%SUBJECT:h_utf8%%";
```

(See “Default PureMessage Policy Script” in the Policy Configuration section of the *Administrator’s Reference* for exact placement.) To edit the policy’s Sieve code directly, open `/opt/pmx6/etc/policy.siv` in a compatible editor.

2. Edit the policy script and, optionally, customize the list of search reasons as described in “Adding and Deleting Custom Log Search Reasons” in the Administrative Groups section of the *Administrators Reference*. Editing the policy adds reason marks to the message log that make it possible to search the log by “Reason”. The second part of “Adding and Deleting Custom Log Search Reasons” describes how to add custom reasons to the **Log Reasons** policy list.
3. Rotate the PureMessage mail and log files so that times in these two logs are synchronized. See “Rotating PureMessage Log Files” for more information.
4. Enable the Log Search Service as described in “Managing the Log Search Index Service” in the *Manager Reference*.

Related concepts

[Default PureMessage Policy Script](#) (page 250)

[Rotating PureMessage Log Files](#) (page 38)

[Managing the Log Search Index Service](#) (page 163)

Related tasks

[Adding and Deleting Custom Log Search Reasons](#) (page 212)

Rotating PureMessage Log Files

PureMessage log files must be periodically cleared to avoid using too much disk space. PureMessage includes a sample configuration file, `logrotate.conf`, that can be used to set the parameters for “rotating” (that is, archiving) the contents of the PureMessage log files.

The `logrotate` utility is a standard component of Linux and is available for other Unix platforms.

Note

This section provides a generic example of using `logrotate`. Alter these instructions as necessary to suit your environment and operating system. Refer to your system's documentation for more information about the `logrotate` utility. Log rotation is a standard maintenance task on Unix systems, and is therefore beyond the scope of PureMessage Support.

`Logrotate` is commonly run as a daily cron job in `/etc/cron.daily`. However, it can be set to run according to whatever frequency suits your installation, or it can be configured to run when logs reach a specified size. The frequency of log rotation depends on the volume of mail your system processes and the amount of logged data you want to preserve (archived data from log files is no longer included in PureMessage's pre-defined reports).

The PureMessage log rotation parameters are determined by the configuration of the `/opt/pmx6/etc/logrotate.conf` configuration file, and the parameters configured for individual PureMessage log files. Parameters for individual log files are stored by default in the `/opt/pmx6/etc/logrotate.d` directory, and include a configuration file for both the system log (`pmx_log`) and the message log (`message_log`), among others. The log configuration files (`pmx_log`, etc) are imported into the (`logrotate.conf`) configuration via the `'include /opt/pmx6/etc/logrotate.d'` statement in the `logrotate.conf` file.

The `logrotate.d` directory contains configuration files for the following logs:

- `access_log`: Records all actions performed via the PureMessage Manager (controls clicked, pages requested, etc). Location: `/opt/pmx6/var/log/manager/`.
- `activity_log`: Records the activity of users and modules associated with the PureMessage Manager. Location: `/opt/pmx6/var/log/`
- `autovac_log`: Records the actions of the `pmx-pg-autovac` program, which is used to maintain the states of database tables in memory. Location: `/opt/pmx6/var/log/`
- `blocklist_log`: If MTA-level policy blocking is enabled, records the IP address of each message processed at the MTA level, along with the associated action. Location: `/opt/pmx6/var/log/`
- `error_log`: Records any errors related to the operation of the PureMessage Manager. Location: `/opt/pmx6/var/log/manager/`
- `message_log`: Records all messages that are processed by PureMessage. Contains both default and user-defined keys. This log is used primarily for reports. Location: `/opt/pmx6/var/log/`
- `pg.log`: Records, by default, any abnormal database activity. Can also be configured to log additional data. Location: `/opt/pmx6/postgres/`
- `pmx_log`: Records a variety of messages issued by `pmx-milter`, including debug, info, notice, warning, and error messages. Location: `/opt/pmx6/var/log/`
- `scheduler_log`: Records the failure of any scheduled job. Optionally, both successes and failures are logged, and failure notifications can be sent to the administrator's email account. Location: `/opt/pmx6/var/log/`
- `vscan_log`: If daemon-mode virus scanning is enabled, records all messages scanned for viruses. Location: `/opt/pmx6/var/log/`

Frequency and backlog settings can be specified either globally in the `logrotate.conf` file or specifically for each log file. Refer to your system's `logrotate` documentation for information about parameters.

Running logrotate as a Scheduled Job

The `logrotate` utility can be run as a scheduled job owned by the PureMessage user (by default, "pmx6"). In the PureMessage Manager, create a new scheduled job. Enter `/path/to/logrotate`

for the command. Add a description, and use the scroll boxes to configure how often the logs will be rotated.

Alternatively, from the command line, create a file named `pmx-logrotate.conf` in `/opt/pmx6/etc/scheduler.d/`, and add an `<event logrotate>` section. For example:

```
<event logrotate>
  desc = "Rotate PureMessage Logs"
  type = exec
  action = '/usr/local/sbin/logrotate /opt/pmx6/etc/logrotate.conf'
  <when>
    s = 0
    m = 0
    h = 3
  </when>
</event>
```

Specifying a State File

On some platforms the default location of the system's state file is not located in a directory for which the PureMessage user has rights. In that case, specify an alternate state file by inserting the following string in the `logrotate` command (before the specification of the `.conf` file):

```
-s /opt/pmx6/var/log/logrotate.state
```

Connecting PureMessage to PostgreSQL

Complete the following steps to ensure proper integration of PostgreSQL with PureMessage:

1. You must explicitly allow connections from any servers in a PureMessage server group by editing the file `/opt/pmx6/postgres/var/data/pg_hba.conf`. Add an entry to this file as follows:

```
host pmx_quarantine pmx 192.168.1.0 255.255.255.0 trust
```

The IP address and mask need to be modified to match your network. If the servers are on different subnets, you can add a separate entry for each one. If the configuration is changed when the PostgreSQL service is already running, you must run `pmx-database restart` as the PureMessage user for the new settings to take effect.

2. On the edge server and central server, edit the `/opt/pmx6/etc/pmx.d/pmdb.conf` configuration file. Set `<host>` to the hostname or IP address of the central server. Also add the following lines:

```
<store pmx_quarantine>
  dsn = 'dbi:Pg:dbname=pmx_quarantine;host=<host>;port=5432'
</store>
```

3. Ensure that `pmx.conf` contains the following lines:

```
!include pmx.d/*.conf
quarantine_type = pmdb
pmx_db = postgres
```

PureMessage includes a default publication (Centralized-Quarantine) that can be used to distribute the quarantine database configuration information to the edge servers that are actively processing and

quarantining messages. See [Managing Publications](#) in the *Manager Reference* for information about using publications.

Configuring Server Groups

A server group is a collection of [edge servers](#) and a central server. Use the “RPC User” account to start and stop services and synchronize files on servers in a PureMessage configuration.

1. On each edge server:
 - a) on the PureMessage Manager’s **Local Services** tab, click **HTTPD (Manager)**.
 - b) On the sidebar, click **RPC User**.
 - c) Configure a password for the RPC User, and click **Save**.
2. On the central server:
 - a) On the sidebar of the **Server Groups** tab, click **Find Hosts**.
A list of hosts is displayed. (Alternatively, manually configure each edge server using the **Add Host** option.)
 - b) In the list of hosts, click the name of an edge server.
This list is only displayed if you selected the **Find Hosts** option in step 4. If you selected **Add Host**, you must manually configure the host information. Click **About this page** on the **Add Host** page to display help for configuring hosts.
 - c) In the **User** field, enter `rpcuser`. In the **Password** field, enter the password you configured for the RPC User on the edge server.
For example, if your edge server is called “server1”, and you configured the RPC User on “server1” to use the password “foobar”, enter “rpcuser” for the username and “foobar” for the password.
 - d) Repeat steps b and c for each additional server.

Testing PureMessage Operations

The `pmx-test` command-line program provides a suite of six tests for checking PureMessage functionality, including performance and general operations testing. Possible tests include: `verify` to test the PureMessage mail server connection (sendmail, Postfix, or Oracle Communications Messaging Exchange Server), `perf` to test PureMessage performance, and `load` to test the PureMessage filter with an incremental load. See `pmx-test` for the syntax and options available with this command-line program.

To test the configuration of a new installation:

1. Run `pmx-test` to test message flow.
2. Run `pmx-qindex` and `pmx-qmeta-index`.
3. Log in to the manager and query the quarantine to ensure the messages are there.
4. Log in to the EUWI as the test user and release messages.
5. Run `pmx-qindex` and `pmx-queue list` to ensure the message is pending release.
6. Run `pmx-qindex` and `pmx-queue list` to ensure the message is pending release.
7. Run `pmx-queue run` and ensure the message is delivered.

Testing PureMessage Policies

It is advisable to test a policy before making it live. Test messages can be sent through PureMessage using the command-line `pmx-test` program, or using the **Test Current Policy** link on the **Policy** tab of the PureMessage Manager. Note that the two methods of testing have different effects on the PureMessage system. See “Testing Policies” in the *Administrator’s Reference* for a description of the differences.

Related concepts

[Testing Policies](#) (page 270)

Related tasks

[Testing the Current Policy](#) (page 112)

Related information

[pmx-test](#)

Configuring Trusted Relays

Trusted relays are mail-filtering hosts that are known to be safe. You can configure a list of trusted relay IP addresses via the **Policy** tab of the PureMessage Manager. PureMessage uses this list to differentiate between trusted and untrusted relays. It is strongly recommended that you configure a list of trusted relay IP addresses to improve spam-filtering performance and reduce *false positives*.

For instructions on configuring trusted relays, see “Configuring Anti-Spam Options” in the Policy section of the *PureMessage Manager Reference*, and “Configuring Spam Detection” in the Policy Configuration section of the *Administrator’s Reference*.

Related concepts

[Configuring Spam Detection](#) (page 275)

Related tasks

[Configuring Anti-Spam Options](#) (page 127)

Configuring autostart for Linux

If you have installed the latest version of PureMessage (version 6.4.1) on a Linux machine using systemd as the system and service manager, you can enable the puremessage.service to start automatically.

For existing PMX installations using systemd as system and service manager:

Run the following script, `pmx_systemd.sh` to enable the `puremessage.service` to start automatically. You need to run the script as root.

```
#!/bin/sh

PREFIX=`su -l pmx6 -c "pmx prefix"`;
INIT=$PREFIX/bin/pmx-init;

cat<<EOT > /etc/systemd/system/puremessage.service
[Unit]
Description=Start and Stop PureMessage
After=syslog.target network.target

[Service]
ExecStart=$INIT start
ExecStop=$INIT stop
Type=oneshot
RemainAfterExit=yes

[Install]
WantedBy=multi-user.target
EOT

if [ $? -ne 0 ];then
    echo "$0 must be run by root";
else
    echo "The PureMessage system startup program $INIT has now been linked
    to systemctl service manager."
fi

systemctl enable puremessage

exit
```

Installing Additional Components

Use the **Install Additional Components** option to add components not previously installed on your server. Additional Components can be added on the command line using the PureMessage Installer.

To install additional components:

1. Run the PureMessage installer at the command line:

```
sh puremessage-XXXX-linux.sh
```

2. On the main menu, press **Tab** to highlight the **Install Additional Components** option, and then press **Enter**.
3. Select additional roles. Select the desired component roles to install. Press **y** for Yes for each desired component. Press **Tab** to highlight **Install these roles**, and then press **Enter**.
4. Answer the series of questions. Follow the instructions in the console window. The questions asked are dependent on the server roles selected.
5. Review entered information. Be sure to review this menu to ensure the desired PureMessage installation instructions are entered. Press **Tab** to select any of the following: **Proceed with the installation**, **Change these install options**, **Return to the main menu**, or **Save the install configuration and exit**. Press **Enter** after highlighting the desired option.

6. Wait for the additional components to install on the server. This menu displays the current status of the PureMessage installation process. The installation may take several minutes. Alternatively, to abort the installation, select **Tab** to **Exit the installer**, **Return to the main menu**, or to **Re-install PureMessage**. Press **Enter** after highlighting the desired option.
7. Review installation log. A window is displayed once PureMessage installation is complete. Sophos recommends reviewing the PureMessage installation log. If not reviewing the log, press **Enter** to select **OK**.
8. Main Menu, exit the installer. Press **Tab** to highlight **Exit**, and then press **Enter**.

Upgrading from a Trial License to a Full License

If you upgrade from a trial license to a full license, you will receive an email with a link to the full license installation script, `license.sh`. When you download this license, it must be saved and run in a directory that is accessible by the “pmx6” user, for example, in `/opt/pmx6/home/`, although it should still be run by the root user.

When you run the script, the installer main menu appears with an option **Upgrade PureMessage License**. Use the down arrow key to highlight this option, and press **Enter** to run it.

To confirm that the license upgrade was successful, log in to the PureMessage Manager, and click the **Support** tab. The license type is displayed in the second column of the **Licensed Components** table. If the license upgrade operation was successful, the license type should be **Full**.

Upgrading PureMessage Components

Use the **Upgrade PureMessage Components** option to check for PureMessage updates, download the updates, and install the updates. Upgrade PureMessage Components on the command line using the PureMessage Installer or via the PureMessage Manager.

Related concepts

[Upgrading PureMessage to Minor Versions](#) (page 62)

[Upgrading to PureMessage Major Versions \(5.6.1 to 6.0\)](#) (page 53)

Uninstalling PureMessage

Use the PureMessage installer to either uninstall specific PureMessage components or to uninstall the entire product.

1. Run the PureMessage installer at the command line:

```
sh puremessage-XXXX-linux.sh
```

2. Select the components to remove. Use this menu to remove any of the following: PureMessage Sendmail, the PureMessage user (pmx6 by default), or the `/opt/pmx6` directory. Press **Tab** to highlight the desired option. Enter **y** (for yes) to select the highlighted option for removal. Press **Tab** to highlight another option. Select **Uninstall these components**, and then press **Enter**.
3. Are you sure? confirmation page. Review the selected components for removal on the confirmation page. To remove the listed components, press **Enter**. To return to the previous page, select **No**, and then press **Enter**.
4. Are you really sure? confirmation page. A second confirmation page is displayed if you chose to remove the PureMessage user. Removing the PureMessage user also removes these components: the pmx mail spool and the pmx home directory (`/opt/pmx6/home`)
To remove the listed components, press **Enter**. To return to the previous page, select **No**, and then press **Enter**.

5. Uninstall confirmation. The selected PureMessage components are removed from the system. The command line prompt is displayed.

The preceding command handles the migration of digest state and global ID mapping data. The remaining migration of end user actions must not be done until the `pmx-qmeta-index` processes performing the reindexing have finished running. After verifying that there are no more `pmx-qmeta-index` processes running on any of the mail processing hosts, run the following command only once on every machine that is running the PostgreSQL database:

```
$ pmx-update-actions
```

Finally, migrate the messages by running the following command:

```
$ pmx-qmeta-index
```

1.3.5 Configuring External Mail Transfer Agents

If you did not select either sendmail or Postfix as the mail transfer agent (MTA) during PureMessage installation, use the instructions below to set up PureMessage to work with an existing MTA. This section describes configuring PureMessage for external versions of sendmail, Sendmail Switch and Postfix.

Note

As external/third party versions of sendmail and Postfix are not quality-assured for integration with PureMessage, Sophos reserves the right not to provide support for an issue that appears to be related to any such custom configuration, and may recommend that you install a version of sendmail or Postfix that is bundled with PureMessage to further a resolution.

If you are using Oracle Communications Messaging Exchange Server, you can configure a direct connection with the PureMessage milter as described in “Configuring PureMessage for Oracle Communications Messaging Exchange Server” in the Sophos Knowledgebase.

Related information

[Configuring PureMessage for Oracle Communications Messaging Exchange Server](#)

Configuring an External Sendmail Installation

PureMessage can be configured to use either an external sendmail or Sendmail Switch installation. ("sendmail" is the open source freeware distribution; "Sendmail Switch" is the commercial version.) See either “Configuring Sendmail” or “Configuring Sendmail Switch” for instructions.

Important

Regardless of whether you are using sendmail or Sendmail Switch, you must configure sendmail aliases for quarantine digests if you intend to use PureMessage’s Quarantine Digest functionality.

Configuring Sendmail

To configure a sendmail installation built from source (as opposed to installing the sendmail distributed with PureMessage), you must first build and install sendmail with milter support, and then connect sendmail with PureMessage.

To configure Sendmail:

1. Build milter support using the method appropriate for your version of sendmail. Sendmail must be built and installed with milter support. For further information on sendmail's milter functionality, see `libmilter/README` in the sendmail source distribution.

- Sendmail versions earlier than 8.12.0

In sendmail versions prior to v8.12.0, milter support was not enabled by default in the sendmail sources. It must be explicitly enabled by adding the following lines to `devtools/Site/site.config.m4` in the sendmail distribution, prior to building sendmail:

```
dnl Milter
APPENDDEF(`conf_sendmail_ENVDEF', `-D_FFR_MILTER=1')
APPENDDEF(`conf_libmilter_ENVDEF', `-D_FFR_MILTER=1')
```

If this file does not exist, create it and then add the lines above.

For sendmail versions prior to v8.12.0, you must add the following line near the beginning of the m4 configuration file (typically `sendmail.mc`) used to generate your `sendmail.cf`. Locate the `VERSIONID` line in `sendmail.mc`, and add the following line right below it:

```
define(`_FFR_MILTER', `1')dnl
```

Note the use of the backtick and apostrophe characters as opening and closing quotes. Ensure that the added text exactly matches the example above.

- Sendmail versions 8.12.0 or later

In sendmail versions v8.12.0 and later, you must enable milter support by adding the following lines to the `devtools/Site/site.config.m4` file in the sendmail distribution, prior to building sendmail:

```
dnl Milter
APPENDDEF(`conf_sendmail_ENVDEF', `-DMILTER=1')
```

Note the use of the backtick and apostrophe characters as opening and closing quotes. Ensure that the added text exactly matches the example above.

2. To compile the sendmail sources, you typically run the following command at the top level of the sendmail source distribution:

```
% sh Build -c
```

3. Copy the sendmail binary to its usual location on your platform, typically `/usr/sbin/sendmail` and/or `/usr/lib/sendmail`, and then generate the sendmail configuration files as described in `sendmail/INSTALL`.

4. Connect sendmail and PureMessage:

- a) Modify `sendmail.mc`

PureMessage milters are defined in the file `pmx.conf`, in the `etc` directory below the PureMessage installation location. For each PureMessage milter you want to set up for processing mail, sendmail must be told how to contact it.

In the directory `sendmail/cf/cf/`, select a configuration file that matches your system configuration. Add a line like the following to `sendmail.mc` before generating `sendmail.cf`:

```
INPUT_MAIL_FILTER(`Policy',
`S=inet:3366@localhost,F=T,T=C:5m;E:8m;R:4m;S:2m')
```


The above assumes PureMessage is listening on port 3366 on the same host where sendmail is running. If PureMessage is running on a different host and port, substitute the hostname for localhost (for example `S=inet:9999@pmxhostname.foo.com`).

The name assigned to the `INPUT_MAIL_FILTER` (`'Policy'` in the example above) is used by sendmail for tagging messages in the syslog. It can be any descriptive text, such as the same name assigned to the PureMessage milter. Milter parameter configuration is further described in the topic “Configuring milter parameters for sendmail” in the Sophos Knowledgebase.

Sendmail calls your milters in the same order specified with `INPUT_MAIL_FILTER` lines.

Sendmail will then contact the filter at the specified port and host every time it handles an SMTP connection.

b) Regenerate sendmail.cf

Regenerate your `sendmail.cf` file from the `sendmail.mc` file. To regenerate your `sendmail.cf` file, run the following commands (replacing the paths to the files appropriately):

```
cd /etc/mail
m4 sendmail_source_dir/cf/m4/cf.m4 /path/to/sendmail.mc >
sendmail.cf
```

Related information

[Configuring Milter Parameters for sendmail](#)

Configuring Sendmail Switch

1. Log into your Sendmail Switch web configuration program.
2. If you don't have an existing configuration, click **New Configuration**. Create a new configuration before continuing. Click **Edit Existing Configuration**.
3. Load the Sendmail Switch configuration file. An example configuration file may resemble the following: `sendmail_switch.m4`.
4. Scroll down to the bottom of the page. On the sidebar, click **Mail Filtering**.
5. Click **Add** to include a new filter.
6. There are two input fields: one is **Filter Name**, the other is **Filter Equates**. Type `Policy` into the **Filter Name** field. Type `S=inet:9999@pmxhostname.foo.com,F=T,T=C:5m;E:8m;R:4m;S:2m` into the **Filter Equates** field. Milter parameter configuration is further described in the topic “Configuring milter parameters for sendmail” in the Sophos Knowledgebase.
7. Click **Apply**.
8. Once the changes are applied, click **Deploy** to update the `sendmail.cf` file.
9. From the lower right side of the page, click **Deploy**.

A confirmation screen with deployment results is displayed.

Related information

[Configuring Milter Parameters for sendmail](#)

Configuring Sendmail Aliases for Quarantine Digests

To use the PureMessage Quarantine Digest function, you must add an alias to the sendmail alias file that matches the address specified in PureMessage. This is specified in the `approve_addr` field of the `pmx-qdigest.conf` configuration file (or on the **Digest Options** page on the **Users** pane of the PureMessage Manager). By default, this address is `pmx-auto-approve@mydomain.com`.

1. Sendmail aliases are stored in the 'aliases' file (usually in the `/etc/mail` directory. Create an entry in the aliases file similar to the following:

```
pmx-auto-approve: |/opt/pmx6/bin/pmx-qdigest-approve
```

2. After adding the alias, run the `newaliases` program (which must be run as root) to regenerate the aliases database file.

Configuring IP Blocking (External Sendmail Version)

PureMessage IP blocking can be configured for an external version of sendmail. To enable IP blocking, you must add an m4 file to your sendmail installation. The required file, `sockmap.m4`, is included in the version of sendmail that is bundled with PureMessage. You must temporarily install PureMessage sendmail, and then copy the `sockmap.m4` file to your existing sendmail installation. Follow the steps in the order they are described below.

Note

As external/third party versions of sendmail are not quality-assured for integration with PureMessage, Sophos reserves the right not to provide support for an issue that appears to be related to any such custom configuration, and may recommend that you install the version of sendmail bundled with PureMessage to further a resolution.

1. Check Sendmail Compilation: You must ensure that your version of sendmail has been compiled with the `SOCKETMAP` option. To check if `SOCKETMAP` was included, as the root user run:

```
sendmail -d0.1 -bt < /dev/null
```

If `SOCKETMAP` is included, it will be displayed among the list of compile options. If, however, `SOCKETMAP` is not there, you will need to recompile sendmail to enable `SOCKETMAP` support. Before you rebuild sendmail, add the following line to the `/devtools/Site/site.config.m4` file:

```
APPENDEF('confMAPDEF', '-DSOCKETMAP')
```

2. Install PureMessage sendmail
 - a) At the command line, as the root user, run:

```
pmx-setup
```

The PureMessage installer is launched.

- b) Select **Install Additional Components**, and press **Enter**.
 - c) Under **Select additional roles that should be installed**, select **Mail Transfer Agent**.
 - d) Select **Install these roles**, and press **Enter**.
 - e) Select **Sendmail**, and then select **Next Question**. Ignore the sendmail configuration prompts, and continue selecting **Next Question** until sendmail has been installed.
 - f) Select **Back to main menu**.
 - g) Select **Exit the installer**.
3. Copy the m4 file to your existing sendmail installation
 - a) Copy `/opt/pmx6/sendmail/cf/feature/sockmap.m4` from the version of PureMessage sendmail that you have just installed to the corresponding directory in your pre-existing sendmail installation.

- b) Add the following line to `$PREFIX/sendmail/etc/mail/sendmail.mc`.

```
FEATURE(`sockmap', `inet:4466@localhost',,)
```

If PureMessage is running on a different host, replace `localhost` with the hostname of the machine on which PureMessage is installed. This hostname must match the one that is specified in `/opt/etc/pmx.d/blocklist.conf` on the server that is running the IP Blocker service.

4. Uninstall PureMessage sendmail
 - a) Select **Uninstall PureMessage Components**.
 - b) Select **PureMessage Sendmail**, and then select **Uninstall these components**.
 - c) If prompted to stop PureMessage, select **Yes**.
 - d) Click **Yes** to confirm the removal. When removal is complete, select **OK**.
 - e) Exit the installer.
5. Update the configuration cache by running `pmx-config --validate-cache`.
6. Start the PureMessage IP Blocker service by running `pmx-blocker start`.
7. Recompile sendmail with the following command:

```
m4 sendmail.mc > /etc/sendmail.cf
```

8. Restart sendmail.
9. At the command line, as the "pmx6" user, run `pmx start` to restart PureMessage.

Note

By default, if sendmail is unable to contact PureMessage's IP Blocker service, the message is passed through. To change this behavior so that messages are tempfailed instead, see the comments in the `sockmap.m4` file.

Configuring an External Postfix Installation

Postfix is the mail transfer agent (MTA) installed during a "Full Install" of PureMessage. You can also configure PureMessage to work with an existing version of Postfix.

While PureMessage will work with any version of Postfix that has content-filtering enabled, older versions of Postfix have security problems. Version 2.1.5 or later is recommended.

The Postfix content-filtering mechanism relies on passing messages via SMTP to PureMessage (the `pmx-milter` program specifically). PureMessage can then modify and/or reinject the message back into Postfix, quarantine the message, or take other actions based on the policy. The `FILTER_README` document included with Postfix provides an overview of the model.

Obtaining Postfix

Source distributions of Postfix are available from <http://www.postfix.org>.

Postfix is also available in various Linux and Unix package formats. Several Linux distributions ship with Postfix. Packages are also available from <http://www.postfix.org/packages.html>.

To get Postfix:

1. Download the current official release and save it to a convenient directory.
 Instructions for compiling and installing Postfix from source are available in the `INSTALL` file contained in the source tarball.

2. Create dedicated “postfix” and “postdrop” user accounts to make Postfix functional. These accounts do not require login access.

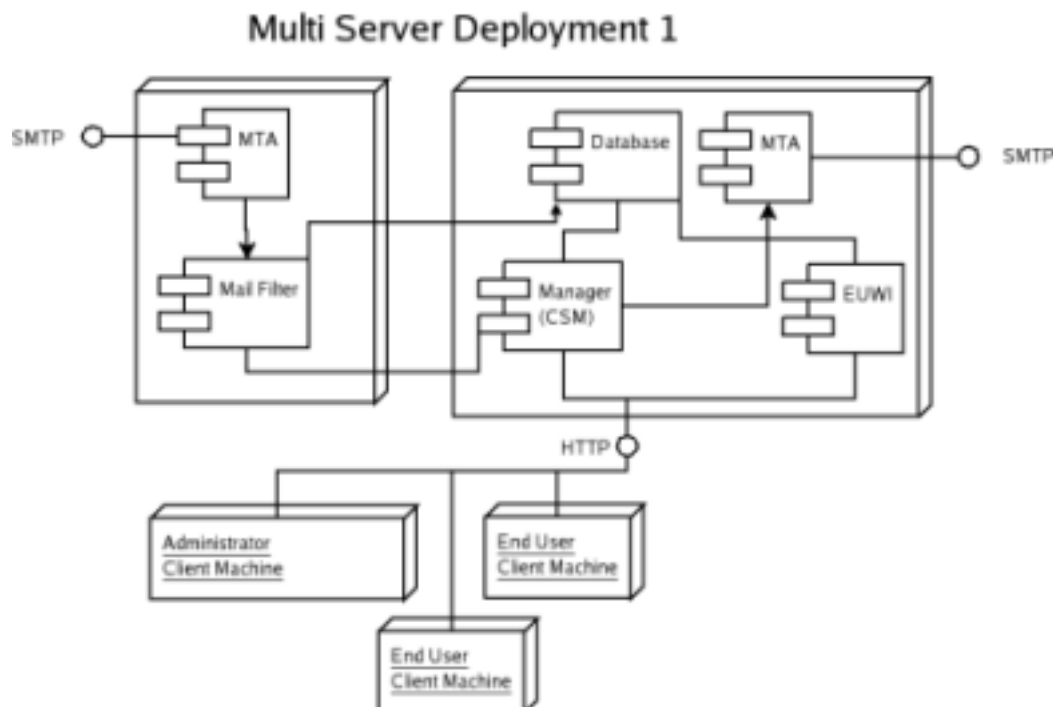
Related information

[Postfix home page](#)

Configuring Postfix

Once installed, you may need to edit `main.cf` to set appropriate values for `myorigin`, `mydestination`, and `mynetworks`. These settings are described in the `main.cf` file. When this is complete, make sure you are able to start Postfix by running `postfix start` and are able to send and receive mail through the system

The following figure (adapted from the Postfix documentation) shows the required setup for filtering email using PureMessage:



To create the pictured message flow:

1. Set Content Filter Port. In `postfix/etc/main.cf`, set `content_filter` to the port PureMessage will be listening on. The default is:

```
content_filter=pmx:127.0.0.1:10025
```

2. Set Interfaces to Non-Postfix Software. In `postfix/etc/master.cf`, add the following:

```
pmx      unix  -      -      n      -      10      smtp

localhost:10026 inet n      -      n      -      10      smtpd
-o content_filter=
-o local_recipient_maps=
-o relay_recipient_maps=
-o myhostname=localhost
-o smtpd_helo_restrictions=
-o smtpd_client_restrictions=
-o smtpd_sender_restrictions=
-o smtpd_recipient_restrictions=permit_mynetworks,reject
-o mynetworks=127.0.0.0/8
```

The `smtp` line defines an interface to use to send messages that have not yet been filtered.

The `smtpd` section defines the port on which to receive messages that have already been filtered.

3. Restart Postfix. Run the `postfix stop` and `postfix start` commands or the `postfix reload` command to make the new settings take effect.

Configuring Postfix to Enable Digest Release by Email

To enable users to release the messages listed in their quarantine digests by email:

1. Open the `postfix/main.cf` for editing.
2. Set the following line:

```
recipient_delimiter=+
```

If this is not done, postfix will look for a username something like "pmx-auto-approve+ffd3b6c7fa2c684f8d6fb1481d7e1297", which usually does not exist.

Configuring Postfix Aliases for Quarantine Digests

To use the PureMessage Quarantine Digest function, you must add an alias to the alias file that matches the address specified in the `approve_addr` field of the `pmx-qdigest.conf` configuration file (or on the **Digest Options** page of the **Quarantine** tab in the PureMessage Manager). By default, this address `pmx-auto-approve@mydomain.com`.

Sendmail aliases are stored in the 'aliases' file (usually in the `postfix/etc/` directory).

To configure Postfix aliases for quarantine digests:

1. Create an entry in the aliases file similar to the following:

```
pmx-auto-approve: |/opt/pmx6/bin/pmx-qdigest-approve
```

2. run the `newaliases` program (which must be run as root) to regenerate the aliases database file.

Configuring IP Blocking (External Postfix Version)

PureMessage IP blocking can be configured for an external version of Postfix.

Note

As external/third party versions of Postfix are not quality-assured for integration with PureMessage, Sophos reserves the right not to provide support for an issue that appears to be related to any such custom configuration, and may recommend that you install the version of Postfix that is bundled with PureMessage to further a resolution.

1. As the root user, add the following line to `<InstallDirectory>/postfix/etc/main.cf`:

```
smtpd_client_restrictions=check_policy_service inet:localhost:4466
```

If PureMessage is running on a different host, replace `localhost` with the hostname of the machine on which PureMessage is installed. The hostname must match the one specified in `/opt/pmx6/etc/pmx.d/blocklist.conf` on the server that is running the IP Blocker service.

2. At the command line, as the PureMessage user (“pmx6” by default), run `pmx-blocker start`.
3. As the root user, restart Postfix.

Although the version of Postfix bundled with PureMessage can be configured to either pass messages through or tempfail messages when the IP Blocker service is unavailable, external Postfix installations can only tempfail messages.

1.3.6 MTA Level IP Blocking

MTA level IP blocking rejects messages originating from IP addresses blacklisted by Sophos Labs. Enabling this option can improve performance by blocking spam before it reaches more complex tests in the policy, such as sender reputation tests.

For installations using PureMessage Sendmail or PureMessage Postfix, configuration of MTA level IP blocking can be done automatically using the PureMessage Manager. Use of other sendmail and Postfix distributions is not supported.

Related tasks

[Enabling or Disabling MTA IP Blocking](#) (page 178)

1.4 Upgrading PureMessage

This document describes how to upgrade to the latest version of PureMessage from a previous version. For a detailed explanation of the various types of PureMessage product updates, see the “PureMessage Versions and Platforms Support Policy”.

Digitally Signed Updates

PureMessage uses certificate validation to verify the authenticity of installations, upgrades, and all subsequent software and data updates. Downloads from Sophos are signed with a private key that matches the public key embedded in the PureMessage validator application. This offers protection in the event that someone attempts to spoof the Sophos website to make it appear as though you are downloading from a trusted source. If a data or software update is interrupted, a partial update is not applied. This ensures the validity and completeness of the download.

In addition, you can configure PureMessage to notify you via email when a download fails the verification. This is done by enabling email notifications for scheduled jobs (see the `scheduler.conf` man page for more information).

Related tasks

[Validating the PureMessage Upgrade Program](#) (page 64)

1.4.1 Upgrading to PureMessage Major Versions (5.6.1 to 6.0)

Important

You must perform this upgrade as the root user. You can only upgrade to PureMessage 6 from PureMessage 5.6.1.

Before starting an upgrade, review the PureMessage “Prerequisites” section to ensure that your system is compatible with PureMessage 6.

This is not a typical upgrade. You will install PureMessage 6 alongside your existing PureMessage 5.6.1 installation by running the `pmx-setup` command.

Optionally, you can then migrate policy settings and other data from PureMessage 5.6.1 to PureMessage 6 by using the PureMessage migration script. For instructions, see “Migrating from Version 5.6.1 to Version 6.”

Prior to installation, you will be prompted to stop Version 5.6.1. For this reason, it is recommended that you perform the upgrade/migration during a time that causes minimal disruption for users.

Once you have completed the installation and migrated all of the necessary 5.6.1 data, you can safely delete the 5.6.1 installation.

Related concepts

[Prerequisites](#) (page 18)

Related tasks

[Migrating from Version 5.6.1 to 6](#) (page 60)

[Upgrading to PureMessage 6 from a Tarball Distribution](#) (page 27)

Single-Server Upgrade (Version 5.6.1 to 6.0)

Important

You must perform this upgrade as the root user.

Perform the following steps to upgrade from PureMessage 5.6.1 to PureMessage 6 on a single PureMessage server. To upgrade a PureMessage deployment of two or more servers, see the “Multi-Server Upgrade (Version 6)” section, which explains the correct order and procedure for updating servers.

1. Run the PureMessage upgrade script, located in `/opt/pmx/bin`:

```
pmx-setup
```

Note

PureMessage uses certificate validation to verify the authenticity of software and data updates. If you want to exercise the utmost caution, you can verify the update program itself. For more information, see “Validating the PureMessage Upgrade Program”.

2. When prompted to retrieve the new version of `pmx-setup`, press **Enter** to accept.

3. You are then prompted to upgrade to PureMessage 6.

Important

You must accept the latest version of PureMessage to take advantage of Version 6 features, and ensure that future updates are performed correctly.

4. Press **Enter** to upgrade to PureMessage 6.
5. A message is displayed that briefly explains the upgrade/migration steps. Press **y** to continue with the Version 6 upgrade.
6. As a precaution, you are prompted to stop PureMessage 5.6.1, even if it is already stopped. Press **Enter**.
7. You are now ready to begin the installation. See “PureMessage 6 Installation” for more information.

Related concepts

[PureMessage 6 Installation](#) (page 57)

Related tasks

[Validating the PureMessage Upgrade Program](#) (page 64)

[Multi-Server Upgrade \(Version 5.6.1 to 6.0\)](#) (page 55)

[Migrating from Version 5.6.1 to 6](#) (page 60)

Multi-Server Upgrade (Version 5.6.1 to 6.0)

Important

You *must* perform this upgrade as the root user.

If you are running PureMessage on multiple servers:

- You *must* upgrade the assigned PureMessage server roles in the correct order (shown below).
- You *must* ensure that you have enough hard disk space available if you are planning to migrate 5.6.1 data (quarantine, logs, reports, etc) after the upgrade. It is recommended that you run the following command as the root user on each PureMessage server to determine the amount of disk space required for installation on that server.

```
du -sh /opt/pmx/
```

- You *must* follow the [single-server upgrade procedure](#) for each server.
- You *must* select the same migration options for each server, if you are using the migration script to import PureMessage 5.6.1 data. For example, if you choose to perform the **Database & Quarantine** step for the server that is running the Database Server role, you must also perform this migration step for all servers in the deployment. For more information, see “Migrating from Version 5.6.1 to 6.”
- You *must* not change the hostname or IP address of any of the servers during the upgrade/migration. This can be done prior to or following the upgrade/migration. Making these changes during an upgrade/migration will result in failures and may leave your system in an unrecoverable state.
- If you have more than one server that is running the Mail Filter role, it is only necessary to stop one mail filter at a time. The other(s) can continue processing messages. Your organization can send and receive mail throughout the upgrade, although access to certain resources (for example, the End User Web Interface and Groups Web Interface) will not be accessible until the entire upgrade process is complete.
- If any of the upgrade steps fail, and you are unsure of how to proceed, contact Sophos Technical Support.

You *must* upgrade any assigned PureMessage server roles in the correct order:

1. Database Server role and Centralized Server Manager (CSM) role
2. Mail Transfer Agent role and Mail Filter role
3. End User Web Interface Server role

To upgrade a multi-server deployment:

1. Upgrade the server that is running the Database Server and Centralized Server Manager (CSM) roles.

Note

Once you begin upgrading the Database server role:

- Mail Filter servers will no longer be able to insert into or read data from the database, and you may see errors in the logs. This is expected and does not affect mail flow.
- Groups Web Interface servers will no longer allow administrators to log in. Thus, administrators should be advised in advance that there will be a service interruption because of the upgrade.
- End User Web Interface servers will no longer allow users to log in. Thus, users should be advised in advance that there will be a service interruption because of the upgrade.

If required, run the migration script on the server running the Database Server and Centralized Server Manager (CSM) roles to import 5.6.1 data immediately following the successful upgrade. For more information, see “Migrating from Version 5.6.1 to 6.”

The database server is now upgraded and migrated. However, the other servers in the deployment will not be able to communicate with the database until they have been upgraded (and, optionally, migrated) to version 6.

2. Start PureMessage on the server that is running the CSM role.
3. One at a time (if there is more than one), upgrade servers that are running the Mail Transfer Agent and Mail Filter roles. If required, run the migration script on each of these servers to import 5.6.1 data immediately following each upgrade.

Note: After the upgrade and optional migrations steps are complete on each server, the individual servers will resume communication with the database and begin to process mail again. *It is highly recommended* that, on the first Mail Transfer Agent and Mail Filter server that is upgraded, you test for correct behavior before proceeding with the upgrade of other servers.

4. One at a time (if there is more than one), upgrade the End User Web Interface servers. If required, run the migration script on each server to import 5.6.1 data immediately following each upgrade.

On each server, after you perform the steps described in the [single-server upgrade procedure](#), you will be prompted to begin the installation on that particular server. For more about multi-server deployment, see the [Custom Installation](#) section.

Related concepts

[Custom Installation \(Version 6\)](#) (page 60)

Related tasks

[Single-Server Upgrade \(Version 5.6.1 to 6.0\)](#) (page 53)

[Migrating from Version 5.6.1 to 6](#) (page 60)

Validating the PureMessage Upgrade Program

Once you have upgraded to PureMessage from a previous version, automatic certificate validation ensures the authenticity of subsequent software and data updates. As an added precaution, before upgrading, you can verify the authenticity of the newest version of the upgrade program, `pmx-setup`.

To validate `pmx-setup`:

1. Run `pmx-setup`. Assuming PureMessage is installed in the default location, at the command-line, enter:

```
/opt/pmx/bin/pmx-setup
```

The `pmx-setup` program detects that a new version is available and prompts you to update.

Important

You must retrieve the new version of `pmx-setup`, otherwise certificate validation will not work.

2. Extract the validator. After `pmx-setup` has updated, you will be returned to the installer. Exit the PureMessage Installer, and, at the command line, enter:

```
/opt/pmx/bin/pmx-setup --check-validator
```

The installer acknowledges that you have chosen to verify the authenticity of the validation tool itself before proceeding with the upgrade. The validation file is extracted.

3. Obtain the `shasum` of the validation file. The `sha1sum` utility is used to calculate and verify SHA1 hashes. If you don't have `shasum`, you must obtain the appropriate version for your operating system. See the `shasum` documentation for specific command syntax.
4. Contact Sophos Technical Support and have a representative confirm that the checksum you have obtained is correct. See “Contacting Sophos” for more information.
5. Verify that your copy of `pmx-setup` is valid. At the command line, run the `pmx-validator` command as shown below:

```
<PathToValidator>/pmx-validator <PathToRepository> <PathTo_pmx-setup>
```

The exact file locations for this command are the `pmx-validator`, `Repository`, and `pmx-setup` entries that are displayed in the installer.

6. Continue with the upgrade. Once verification is complete, enter:

```
/opt/pmx/bin/pmx-setup
```

Related concepts

[Contacting Sophos](#) (page 74)

PureMessage 6 Installation

When upgrading from PureMessage 5.6.1 to PureMessage 6, you will be prompted to install version 6.

The **Full Installation** option installs PureMessage on a single server. The **Custom Installation** option allows you to upgrade to version 6 in a multi-server deployment.

If you want to make changes to a multi-server deployment, such as using a different mail transfer agent or re-assigning PureMessage roles, you should re-install PureMessage. For more information, see [Custom Installation](#) in the “PureMessage Installation” section.

Note

While many terminal applications and environment variables will work with the PureMessage installer, Sophos has tested and supports using `xterm`, `putty` and the console with a `$PMX_TERM` environment variable of either `xterm` or `vt100`. If you have trouble with your installer's display, use one of these combinations.

PureMessage includes a menu-based installer that streamlines the installation process. This installer launches in a console window. The window is partitioned into two sections; the left side displays

installation options, and the right side displays help text. Use the following keyboard commands to navigate the installer:

- **Tab** : moves cursor from one option to another
- **Left/Right Arrow** : shifts focus between the left and right panes of the installer
- **Up/Down Arrow** : scrolls the text in the right pane of the installer (if the right pane has focus)
- **Space Bar** : selects the highlighted option or check box(es)
- **Enter** : accepts the selected options or check box(es)

Important

Sophos strongly recommends that you do not resize the installer console window once the PureMessage installer launches. Resizing the console window may cause the installer to terminate. Before installing, stop any process that uses port 25, including existing versions of Postfix or sendmail. Regardless of whether you configure PureMessage to work with an existing mail transfer agent (MTA) or the sendmail or Postfix version that is bundled with PureMessage, you must start the designated MTA after installation is complete.

Related concepts

[Custom Installation](#) (page 31)

[Prerequisites](#) (page 18)

Full Installation (Version 6)

The **Full Install** option installs a complete PureMessage system on a single server. To finish your upgrade by installing PureMessage on two or more servers, see the “Custom Installation (Version 6)” section.

A full installation consists of the following:

- Centralized Server Manager (CSM)
- Database server
- End User Web Interface/Groups Web Interface
- Mail filtering
- PureMessage Postfix

Important

If you will be migrating data from your Version 5.61 installation (see “Migrating from Version 5.6.1 to 6”), it is *not* recommended that you modify the PureMessage 6 installation until the migration is complete. Migrating data overwrites core configuration settings in PureMessage 6 with settings from PureMessage 5.6.1.

Follow the steps below to complete a **Full Install**:

1. Welcome to PureMessage. Proceed with installation? To continue, press **Enter** .
2. Enter a directory for the PureMessage installation. Select the directory where PureMessage and its supporting files will be installed. The default location is `/opt/pm6/`. The directory must have at least 500 MB of free space. Additional space is required for log files and messages stored in the PureMessage quarantine. Type a location at the prompt and press **Enter** , or press **Enter** to select the default installation location.

Important

It is not recommended that you move any of the directories that are located directly beneath `/opt/pmx6/` in the PureMessage installation directory. Directories such as `postfix`, `postgres`, etc contain binary files and libraries that are required for PureMessage updates. Moving or symlinking these directories could cause upgrades to fail.

3. Accept the Sophos End User License Agreement. Use the **Page Up** and **Page Down** keys to view the entire user agreement. Be sure not to resize the console window. Press **Tab** to move the cursor to the **I accept the licensing terms** box. Press **Space Bar** to select this option. Press **Tab** to select **OK**, and then press **Enter**.
4. Select the Full Install option. Press **Tab** to select **Full Install**, and then press **Enter**. The installer will guide you through the remainder of the installation process. After completing each step, use the **Tab** key to highlight **<Next Question>**, and press **Enter**. At any point in the process, you can return to the previous question or go back to the main installation menu. Refer to the documentation in the right pane of the installer for information about the currently displayed installation option.
5. Complete the installation. Once you have finished entering the installation information, you are prompted to **Proceed with the installation**. Press **Enter**.
6. Exit the PureMessage Installer. Once you have completed the steps, exit the installer.
7. Migrate data? You are prompted to run the PureMessage migration script. If you want to migrate policy settings and other data, see “Migrating from Version 5.6.1 to 6.” If you prefer not to migrate data, or you want to migrate manually, proceed to the next step.
8. Start PureMessage. At the command line, as the PureMessage user (by default, 'pmx6'), run:

```
pmx start
```

9. Start the mail transfer agent. Once you have completed the installation and started PureMessage, you must start the mail transfer agent (MTA) that will be used with PureMessage. You can start the MTA in the PureMessage Manager or from the command line.
 - Manager: On the **Local Services** tab, under **Background Services**, select the check box next to the SMTP service and click **Start**.
 - Command Line: As the 'pmx6' user, run `pmx-service start smtp`.

If you want to use PureMessage with an existing mail transfer agent, see “Configuring External Mail Transfer Agents” in the Post-Installation Procedures section of the *Getting Started Guide*.

Note

During installation, PureMessage automatically adjusts the PostgreSQL database in an attempt to improve performance. If the adjustments are successful, PureMessage displays a confirmation message. Depending on the shared memory configuration of your system, however, additional manual steps may be required.

If your system settings are not within the recommended range, PureMessage tries to make the necessary adjustments. In some cases, the updated settings are saved to file named `postgresql.conf.recommended`. You can accept these settings by renaming the file to `postgresql.conf`. If your system does not meet even the minimum requirement for shared memory, an error message is displayed. For more information, see “Tuning PostgreSQL for PureMessage” in the Sophos Knowledgebase.

Related concepts

[Custom Installation](#) (page 31)

[Configuring External Mail Transfer Agents](#) (page 45)

Related tasks

[Migrating from Version 5.6.1 to 6](#) (page 60)

[Validating the Installation Script](#) (page 36)

Related information

[Tuning PostgreSQL for PureMessage](#)

Custom Installation (Version 6)

Use the **Custom Installation** option to install PureMessage 6 on two or more servers.

These instructions assume that you are maintaining the same configuration used in version 5.6.1. If you want to re-assign [roles](#) or use a different mail transfer agent, it is recommended that you re-install PureMessage 6 on each server.

When upgrading a PureMessage deployment of two or more servers, retain the role(s) currently assigned to each server. For instance, if the server you are upgrading is running the **Centralized Server Manager** and **Database Server** roles, select, **Custom Install**, and then select the same roles.

For more information, see “Custom Installation” in the “PureMessage Installation” section.

Related concepts

[Custom Installation](#) (page 31)

[Configuring an External Sendmail Installation](#) (page 45)

[Configuring an External Postfix Installation](#) (page 49)

Related information

[Tuning PostgreSQL for PureMessage](#)

Migrating from Version 5.6.1 to 6

Important

- Migrating data overwrites core configuration settings in PureMessage 6 with settings from PureMessage 5.6.1. It is therefore *not* recommended that you modify the PureMessage 6 installation until the migration is complete.
- When upgrading servers in a multi-server deployment, you must migrate servers in the correct order, and then run the migration script on each server, immediately after each one is upgraded. See “Migrating a Multi-Server Deployment” below.

If have finished installing PureMessage 6 alongside an existing 5.6.1 installation, you have the option of migrating policy settings and other data from your 5.6.1 installation to the new installation directory. Immediately following installation of Version 6, you are prompted to run the migration script.

Note

You can only migrate data from the most recent version of PureMessage (5.6.1). If you are running an older version of PureMessage, upgrade to Version 5.6.1 before migrating.

The migration is done by way of a command-line tool (`pmx-v6-migrate`), which leads you through the steps in the required order.

Sophos Professional Services modules and settings are not imported as part of the migration. Contact Professional Services if you require any assistance. Although core configuration files are migrated, there are some configuration files that are not imported to version 6. For detailed results of the migration, see `/opt/pmx6/var/log/migrate_pmxv5.log`.

If you prefer to migrate data manually, or you choose not to migrate any data, do not perform these steps.

To migrate your data on a single-server configuration.

1. As the PureMessage user (by default, 'pmx6'), run the PureMessage migration script located in `/opt/pmx6/bin`:

```
pmx-v6-migrate
```

You are presented with three migration options, which must be run in order shown.

Note

The optional Step 3 (**Database & Quarantine**) may take a very long time, depending on the size of the quarantine and your data.

2. At the **Enter selection** prompt, type **1**, and press **Enter**. When prompted to overwrite your current PureMessage 6 configuration settings, type **y**.
The results are displayed at the command line as the configuration files are migrated.
3. Press **Enter** to return to the migration menu.
4. At the **Enter selection** prompt, type **2**, and press **Enter**. When prompted to overwrite your current PureMessage 6 log search files and data, type **y**.
5. Press **Enter** to return to the migration menu.
6. [Optional] At the **Enter selection** prompt, type **3**, and press **Enter**. Carefully read the message displayed at the command prompt. If you are ready to proceed, type **y**.
The results of the migration are displayed.
7. At the **Enter selection** prompt, type **4**, and press **Enter** to exit the migration script.

Note

Migrating a Multi-Server Deployment

- Migrate data for each server separately.
- Upgrade each server individually, running the migration script on each server before proceeding to the next server.
- Select the same migration options for each server. For example, if you choose to perform the **Database & Quarantine** step for the server that is running the Database Server role, you must also perform this migration step for all servers in the deployment.
- Before migrating data, start the server that is running the Database Server and Centralized Server Manager (CSM) roles.
- When migrating on multihomed servers, the IP address selected during the version 6 upgrade must match the IP address that is specified in `etc/location` for version 5.6.1.

For more information, see “Multi-Server Upgrade (Version 6).”

Related concepts

[Installing PureMessage](#) (page 18)

Related tasks

[Multi-Server Upgrade \(Version 5.6.1 to 6.0\)](#) (page 55)

[Validating the PureMessage Upgrade Program](#) (page 64)

1.4.2 Upgrading PureMessage to Minor Versions

Important

You must perform all upgrades as the root user. You can only upgrade to the latest version of PureMessage from a supported version of PureMessage. "For information about currently supported versions, see the "PureMessage Versions and Platform Support Policy".

PureMessage includes a menu-based installer that streamlines the upgrade process. This installer launches in a console window. The window is partitioned into two sections; the left side displays installation/upgrade options, and the right side displays help text. Use the following keyboard commands to navigate the installer:

- **Tab** : moves cursor from one option to another
- **Left/Right Arrow** : shifts focus between the left and right panes of the installer
- **Up/Down Arrow** : scrolls the text in the right pane of the installer (if the right pane has focus)
- **Space Bar** : selects the highlighted option or check box(es)
- **Enter** : accepts the selected options or check box(es)

Important

Sophos strongly recommends that you do not resize the installer console window once it launches. Resizing the console window may cause the installer to terminate.

Use the **Upgrade PureMessage Components** option in the PureMessage installer to check for PureMessage updates, download the updates, and install the updates.

Note

During upgrades, PureMessage automatically adjusts the PostgreSQL database in an attempt to improve performance. If the adjustments are successful, PureMessage displays a confirmation message. If the adjustments fail, it is likely due to the shared memory configuration of your system. PureMessage will revert the changes and create a `postgresql.conf.recommended` file that contains appropriate system settings. For more information, see "Tuning PostgreSQL for PureMessage" in the Sophos Knowledgebase.

Related tasks

[Upgrading to PureMessage 5.6.1 from a Tarball Distribution](#) (page 26)

Related information

[Tuning PostgreSQL for PureMessage](#)

Single-Server Upgrade

Important

You must perform this upgrade as the root user.

Perform the following steps to upgrade to the latest version of PureMessage on a single-server configuration:

1. Run the PureMessage installation script located in `/opt/pmx6/bin`:

```
pmx-setup
```

If prompted to upgrade to the latest version of `pmx-setup`, press **Enter** to retrieve the new version.

Important

You must accept the latest version of `pmx-setup` to ensure that updates continue to be performed correctly.

Note

PureMessage uses certificate validation to verify the authenticity of software and data updates. If you want to exercise the utmost caution, you can verify the update program itself. For more information, see “Validating the PureMessage Upgrade Program”.

2. On the main menu, select the **Upgrade PureMessage Components** option, and then press **Enter**.
3. Select **Check for updates**, and then press **Enter**. The installer queries for new updates.
4. If updates are found, select **Upgrade components**, and then press **Enter**.
5. You are prompted to stop PureMessage before continuing the installation. Select **Yes**, and press **Enter**.

Updating the PureMessage components may take a few minutes. When all packages are upgraded successfully, press **Enter**.

6. Select **Return to the main menu**, and then press **Enter**.
7. Select **Exit the installer**, and then press **Enter**.

Important

It is not recommended that you move any of the directories that are located directly beneath `/opt/pmx/` in the PureMessage installation directory. Directories such as `postfix`, `postgres`, etc contain binary files and libraries that are required for PureMessage updates. Moving or symlinking these directories could cause upgrades to fail.

Related tasks

[Validating the PureMessage Upgrade Program](#) (page 64)

Multi-Server Upgrade

Important

You must perform this upgrade as the root user.

If you are running PureMessage on multiple servers, you must follow the [single-server upgrade procedure](#) on each server. You *must* upgrade PureMessage server roles in the correct order (shown below):

1. Database Server role and Centralized Server Manager (CSM) role
2. Mail Transfer Agent role and Mail Filter role
3. End User Web Interface Server role

Important

It is not recommended that you move any of the directories that are located directly beneath `/opt/pmx6/` in the PureMessage installation directory. Directories such as `postfix`, `postgres`, etc contain binary files and libraries that are required for PureMessage updates. Moving or symlinking these directories could cause upgrades to fail.

Related tasks

[Single-Server Upgrade](#) (page 63)

Validating the PureMessage Upgrade Program

Once you have upgraded to PureMessage from a previous version, automatic certificate validation ensures the authenticity of subsequent software and data updates. As an added precaution, before upgrading, you can verify the authenticity of the newest version of the upgrade program, `pmx-setup`.

To validate `pmx-setup`:

1. Run `pmx-setup`. Assuming that PureMessage is installed in the default location, at the command line, enter:

```
/opt/pmx6/bin/pmx-setup
```

The `pmx-setup` program detects that a new version is available and prompts you to update.

Important

You must retrieve the new version of `pmx-setup`, otherwise certificate validation will not work.

2. Extract the validator. After `pmx-setup` has updated, you will be returned to the installer. Exit the PureMessage Installer, and, at the command line, enter:

```
pmx-setup --check-validator
```

The installer acknowledges that you have chosen to verify the authenticity of the validation tool itself before proceeding with the upgrade. The validation file is extracted.

3. Obtain the `sha1sum` of the validation file. The `sha1sum` utility is used to calculate and verify SHA1 hashes. If you don't have `sha1sum`, you must obtain the appropriate version for your operating system. See the `sha1sum` documentation for specific command syntax.
4. Contact Sophos support and have a representative confirm that the checksum you have obtained is correct. See “Contacting Sophos” for more information.
5. Verify that your copy of `pmx-setup` is valid. At the command line, run the `pmx-validator` command as shown below:

```
<PathToValidator>/pmx-validator <PathToRepository> <PathTo_pmx-setup>
```

The exact file locations for this command are the `pmx-validator`, `Repository`, and `pmx-setup` entries that are displayed in the installer.

6. Continue with the upgrade. Once verification is complete, enter:

```
pmx-setup
```

Related concepts

[Contacting Sophos](#) (page 74)

1.5 Quick Reference Guide

Overview

PureMessage is an email-filtering program that analyzes email messages at the network gateway. PureMessage checks messages for characteristics that indicate “spam” (unsolicited bulk email), scans messages for viruses in conjunction with Sophos Anti-Virus, and can also check messages to ensure that corporate email adheres to an organization's communication [policy](#).

PureMessage processes mail in three main steps. (Functionality may vary, depending on the configuration.)

1. The mail server passes a message to PureMessage: The mail server (sendmail, Postfix, Oracle Communications Messaging Exchange Server) passes email to PureMessage via PureMessage's [milter](#) service.
2. PureMessage applies the policy filter to the message: The policy script contains tests that check for viruses, spam indicators, or other message characteristics. The tests have associated actions that determine what happens to the message if the test returns true. Available actions include:
 - **Delivery:** Delivery actions can be configured to either deliver the message to the original recipient, redirect the message to other recipients, or remove and add recipients. The policy script can be configured to alter the original message, for example, by altering the subject line or adding a message header.
 - **Quarantine:** The policy script can be configured to quarantine the message if certain conditions are met (such as triggering a sufficient number of anti-spam rules). Quarantined messages are copied to the quarantine directory. Optionally, the original recipient may be notified that the message was quarantined via a quarantine digest.
3. Quarantined messages are administered: End users can release desired messages from the quarantine by responding to quarantine digests or via the [End User Web Interface](#) (EUWI). Other messages are automatically archived or deleted from the quarantine using the `pmx-qexpire` program, which is configured by default during the installation to run as a scheduled job.

This document provides an overview of PureMessage. It introduces the major components of a PureMessage installation and explains how they work together to keep networks spam-free. For details about installing PureMessage, see the “Installation Guide”. For complete operation and configuration instructions, see the *Manager Reference* and the *Administrator’s Reference*.

Related concepts

[Installing PureMessage](#) (page 18)

1.5.1 Starting and Stopping PureMessage

After installing PureMessage and configuring the connection to the mail server (as described in the Installation Guide, PureMessage services are started using the `pmx` program. The `pmx` program must be run as the PureMessage user by default, “pmx6”). The password for the “pmx6” user is set during installation.

- To start PureMessage services:
 - a) As the 'pmx6' user, enter the following at the command line:

```
pmx start
```

- To stop PureMessage services:
 - a) As the 'pmx6' user, enter the following at the command line:

```
pmx stop
```

For other command-line options, refer to the *Administrator’s Reference*.

Related concepts

[Installing PureMessage](#) (page 18)

Related information

[pmx - the PureMessage control program](#)

1.5.2 PureMessage and Mail Servers

The PureMessage distribution includes both the Postfix and sendmail SMTP mail servers. See the Installation Guide for more information. In addition, PureMessage can integrate with existing Postfix, sendmail, Sendmail Switch, and JSMS installations.

Postfix

PureMessage can be configured to work with the Postfix mail server. A distribution of Postfix is included with PureMessage, and it is the the MTA installed by default during a “Full Installation” of PureMessage. Alternatively, PureMessage can be configured to work with an existing Postfix installation. PureMessage communicates with the policy engine in PureMessage by using PureMessage as an SMTP proxy. See “Configuring an External Postfix Installation” in the Installation Guide for more information.

Sendmail

PureMessage works in conjunction with either the open source version of sendmail or the commercial version, Sendmail Switch. A distribution of sendmail is included with PureMessage. Alternatively, PureMessage can be configured to work with an existing sendmail or Sendmail Switch installation. PureMessage communicates with sendmail via the Milster (Policy) service. Refer to the Installation Guide for information on installing the PureMessage distribution of sendmail, or configuring PureMessage to work with an existing sendmail or Sendmail Switch installation.

If you are using the distribution of sendmail included with PureMessage, the sendmail documentation is available via the “Documentation for Related Applications” section of the *Administrator's Reference*. Otherwise, refer to the sendmail or Sendmail Switch documentation included with your installation.

Oracle Communications Messaging Exchange Server

PureMessage can be configured to work with the Oracle Communications Messaging Exchange Server mail transfer agent. This MTA is supported on Solaris only.

Related concepts

[Installing PureMessage](#) (page 18)

[Configuring an External Postfix Installation](#) (page 49)

[PureMessage Services](#) (page 67)

[Documentation for Related Applications](#) (page 5)

Related information

[Configuring PureMessage for Oracle Communications Messaging Exchange Server](#)

1.5.3 PureMessage Services

PureMessage operates as a series of services. Background services such as HTTPD (Manager) and Milster (Policy) are activated when you start PureMessage. PureMessage also uses a set of scheduled jobs to perform a variety of administrative tasks at specified times. These jobs are controlled collectively by the [Scheduler](#), which itself is a background service. The services are:

- **AntiVirus Service:** The Sophos virus-scanning service. This service is enabled only when the **Run As Service** check box is selected in on the **Anti-Virus Options** page of the **Policy** tab.
- **HTTPD (RPC/UI):** The web server and RPC services that make up the [End User Web Interface](#) and the [Groups Web Interface](#).
- **IP Blocker Service:** The PureMessage service that runs (if MTA level IP blocking is enabled). This blocks messages originating from IP addresses blacklisted by Sophos Labs and from senders that are otherwise deemed to be spammers, which can improve performance by blocking spam before it undergoes more complex testing via the PureMessage [policy](#).
- **HTTPD (Manager):** The web-based administrative interface to PureMessage.
- **Milster (Policy):** The PureMessage component that interacts with sendmail or Postfix and runs the policy engine.
- **PostgreSQL Service:** The default relational database back end used for reporting and the PureMessage [quarantine](#).
- **Queue Runner Service:** The service that manages and flushes the PureMessage mail queue.
- **Scheduler Service:** PureMessage relies on the execution of commands at specified intervals, which is handled by the PureMessage Scheduler Service.

- Mail Transfer Agent (sendmail, Postfix, Oracle Communications Messaging Exchange Server): The service that runs the configured mail transfer agent.

Scheduled Jobs

Scheduled jobs, enabled and disabled collectively by the Scheduler Service, are PureMessage programs that run at set times to accomplish specific tasks. Preconfigured scheduled jobs are displayed in the Manager, and many are enabled by default. Clicking on the name of an individual job displays configurable time settings for that job. You can also add other PureMessage programs to the set of scheduled jobs.

Related concepts

[Managing the IP Blocker Service](#) (page 164)

[Documentation for Related Applications](#) (page 5)

1.5.4 PureMessage Manager

The Manager is a web-based graphical user interface for managing and configuring PureMessage. Installed as part of each PureMessage *role* during installation, the Manager runs as the HTTPD (Manager) service.

The Manager uses Secure Sockets Layer (SSL) and is available through port 18080. To access the Manager, use the following URL, where <hostname> is the name of the host on which PureMessage is installed, and <domain> is your organization's domain:

```
https://<hostname>.<domain>:18080/
```

The Manager interface is organized into tabs that correspond to the key areas of PureMessage functionality:

- Dashboard Tab: Offers a high-level diagnostic overview of the PureMessage system.
- Policy Tab: Provides access to PureMessage policy rule configuration and information. The default page for this tab is the **Rules** page, which provides a graphical interface to the Sieve-based PureMessage policy. Click **edit source** to view and edit the Sieve code directly. This tab also displays all of the configured policy lists and maps. Lists and maps can be added, configured and tested from this tab.
- Quarantine Tab: Provides information and management capabilities for the PureMessage quarantine, a temporary storage area for problem messages. Typically, quarantines are used for virus-infected messages that cannot be safely deleted automatically, or messages awaiting review by administrators or end users to determine if they are spam. This tab also has settings for quarantine digest rules and options, and End User Web Interface configuration features.
- Reports Tab: A summary of essential reports data, including virus and spam statistics and system information. For each report, you can modify the graph format, report period, and the server(s) included. You can also export the report data to comma-separated values (CSV) format and configure an automated mailout for each report.
- Local Services Tab: Provides summary information and management for PureMessage services.
- Server Groups Tab: View the status of servers and manage server groups in multi-server installations.
- Support Tab: Contains information about PureMessage licenses and support arrangements, as well as links to support resources and several general PureMessage management utilities.

For information about using the PureMessage Manager, see the *Manager Reference*.

1.5.5 PureMessage Command-Line Interface

PureMessage tasks can be performed at the command line using utility programs and configuration files. For example, `pmx-policy` is the command-line interface to the PureMessage policy engine; `pmx-qman` is used to access quarantined messages. The `sophos.conf` configuration file is used to specify options for the Sophos Anti-Virus engine. See the relevant section of the *Administrator's Reference* for descriptions of these programs and hyperlinks to documentation for the individual programs.

“pmx” User

A PureMessage (“pmx6”) user is created by default during installation. The PureMessage user’s home directory is located beneath the PureMessage installation directory (by default, `/opt/pmx6`). The PureMessage user is configured to use a Bourne-compatible shell; this configuration must not be altered. The PureMessage user account runs PureMessage programs. You are prompted to specify a password for the PureMessage user during installation.

1.5.6 Policy Engine

Policies, which are built using the [Sieve](#) language, provide the filtering definition for PureMessage. PureMessage can be configured to use multiple policy scripts but uses only one by default. The default policy script is stored at `/opt/pmx6/etc/policy.siv`, beneath the PureMessage installation directory.

Policies consist of rules; rules consist of tests and actions. As the policy processes messages, rules are executed on the message in the order of their configuration.

Configure policies on the **Policy** tab of the PureMessage Manager, or by using the `pmx-policy` command-line program. A default policy is installed and enabled during PureMessage installation. The default policy varies according to your PureMessage license; for example, if you do not have a license for the PureMessage Virus component, virus-checking rules are not configured.

Tests define the characteristics of the message that must be matched in order for the action to be executed. Multiple conditions can be configured for a single rule.

Actions are the parts of rules that determine what happens to a message that meets or does not meet a test. Multiple actions can be specified.

You can modify the PureMessage policy using the options available on the **Policy** tab of the Manager. Clicking a rule description launches a new Rules page with configurable fields. Alternatively, clicking **see the source** displays the Sieve code that underlies the policy interface. The code can be edited directly through the Manager. See the *Manager Reference* and *Administrator's Reference* for instructions on editing the policy.

PureMessage has a Policy Repository for storing Sieve snippets and scripts that can be incorporated into a PureMessage policy script. By default, the repository contains a number of general-purpose snippets that can be copied and pasted from the Policy Repository to the PureMessage policy and vice versa.

Related concepts

[About Tests](#) (page 81)

[About Actions](#) (page 93)

Related information

[pmx-policy](#)

About PureMessage Mail Filtering

The PureMessage mail filter (`pmx-milter`) integrates with the mail transfer agent (MTA) layer of your email infrastructure (sendmail, Postfix, or Oracle Communications Messaging Exchange Server) to process messages. Filtering options are defined using a [Sieve](#)-based policy script (`policy.siv`) which specifies the order of various tests and actions to be performed on each message. The following list describes some of the configurable filtering tasks:

- Check message content and attachments for known viruses and take specified actions.
- Check messages for suspicious attachment types (such as executables) and take specified actions.
- Identify messages that are spam and take specified actions.
- Check messages for keywords or phrases and take specified actions.
- Log and archive messages based on selective criteria.
- Remove, replace, or add recipients based on selective criteria.
- Add specific banners in the header or body of outgoing messages.
- Annotate email message headers (for example, Subject) with various kinds of additional information.

Tasks can be applied differently to different users and groups within your organization, and can be flexibly ordered and combined to execute your desired email policy.

Related information

[pmx-milter](#)

Lists and Maps

Lists are used to store groups of domains, addresses or other unique identifiers referenced by particular policy rules. For example, in the default PureMessage policy, the first rule checks the message relay against the relays configured in the Internal Hosts list.

Address maps are used to associate one email address with another, either for the purpose of redirecting notifications generated by PureMessage (such as emailed digests of quarantined mail), or for the purpose of assigning one user's email preferences to other accounts (for End User Web Interface usage).

Note: Lists and address maps with more than 5,000 entries should be converted into CDB format for better performance. See "CDB Lists and Maps" in the Policy Configuration section of the *Administrator's Reference* for instructions on editing `lists.conf` and converting the files with `pmx-makemap`.

Related tasks

[CDB Lists and Maps](#) (page 254)

Message-Handling Options

"Message-handling" describes the actions that PureMessage performs on messages and depends on the type of message being handled.

Message handling also depends on the components that are configured in your PureMessage installation. For example, the [End User Web Interface](#) provides individual user access to PureMessage

mail-filtering functionality via a client browser. Installing this optional PureMessage component lets end users manage their own quarantined messages, whitelists and blacklists, and configure individual mail-filtering options.

Message handling is implemented via the policy script, which is described in detail in the [Policy Configuration](#) section of the Administrator's Reference. The sections below provide a general overview of the actions that can be specified in the policy script.

Spam-Handling Options

The mechanism by which spam probabilities are calculated, and the actions associated with probability thresholds, are described in the [Tuning Spam Detection](#) section.

- **Tag and Deliver:** Alter the "Subject" header of messages delivered to end users, generally if the message's spam probability is over a certain "threshold". For example, the default policy alters the Subject header if the message's spam probability is greater than 50%. Users can implement filters within their email client based on the subject line modification, as described in the [End User Management](#) section of the Administrator's Reference.
- **Silently Tag and Deliver:** Add a custom message header that records the spam probability and the anti-spam rules that were triggered by the message. The default policy adds an `X-PerlMx-Spam` header to all messages with a spam probability. Custom headers are not visible in most client email programs.
- **Quarantine:** Quarantine messages with a spam probability greater than a specified percentage, allowing administrators to review quarantined messages. It can be used in conjunction with the options described above. By default, the PureMessage policy copies messages with a probability of 50% or greater to the quarantine.
- **Quarantine and Digest:** Move messages above a certain probability to the quarantine. [Quarantine Digests](#) inform users of which messages have been quarantined.

Virus-Handling Options

The following virus-handling options are available:

- **Clean the Message:** When the anti-virus engine detects a virus in a message, PureMessage can attempt remove the virus from the message. If the virus cannot be cleaned from the message, PureMessage removes the infected attachment and generates and generates a notification message for the recipient, based on a template. This is the default behavior in the policy for messages that originate from external hosts.
- **Reject or Discard the Message:** PureMessage can either reject or discard messages containing viruses. By default, messages originating from internal hosts are rejected.
- **Scanfailed Action:** When a message contains content that cannot be scanned, such as an encrypted attachment, PureMessage can take specific action, based on the virus engine's response to the failed scan.

Policy-Handling Options

In this context, "policy-handling options" refer to tests and actions that can be implemented in the policy script. These options can include checking for attachments, adding banners to messages, or placing the result of shell commands in a template variable. Because policy-handling options vary depending on an organization's needs, there are no general recommendations.

For examples on configuring policy-handling options, see [Customizing Policies](#) in the Administrator's Reference.

1.5.7 Quarantine

The PureMessage quarantine stores messages that have been quarantined according to the configuration of the PureMessage policy. For example, messages quarantined because their spam probability exceeds a certain level.

By default, PureMessage uses the PostgreSQL database to index its quarantine.

Messages are quarantined when the **Copy the message to quarantine** (“`pmx_file`”) or **Quarantine the message** (“`pmx_quarantine`”) action is used in a policy rule.

When a message is quarantined, the message is moved to the `var/qdir/new` directory beneath the PureMessage installation directory. The quarantine indexer then assigns a quarantine identifier to the message, updates index files and moves it to the `var/qdir/cur` directory.

When messages are first quarantined, they are temporarily stored in `/opt/var/qdir/new`. At regular intervals, the quarantine indexer parses this directory, assigns quarantine ID numbers to the messages, updates the index, and moves the messages to `/opt/var/qdir/cur`.

Quarantined messages are managed using either the **Manage Quarantine** feature in the PureMessage Manager or the `pmx-qman` command-line program. Message management actions include deleting, approving for delivery, and saving.

Quarantine Digests

When quarantine digests are enabled, end users receive email notifications of quarantined messages. Notifications contain lists of quarantined messages that were originally destined for particular users. Users can reply to automatically release the desired messages from the quarantine.

Centralized and Consolidated Quarantines

Administrators can configure PureMessage to use either a centralized or consolidated quarantine. In a centralized quarantine, metadata from multiple quarantines is collected in a single PostgreSQL database. In a consolidated quarantine, messages are drawn from multiple quarantines and stored in a single location.

Related concepts

[About Actions](#) (page 93)

[Digests Management](#) (page 286)

[Consolidated vs. Centralized Quarantines](#) (page 284)

1.5.8 End User Web Interface

The End User Web Interface (EUWI) gives PureMessage users access to email-filtering features. Users can view and manage messages that are quarantined by PureMessage, manage user-specific sender lists, and configure various email-filtering options.

The EUWI is a PureMessage service that runs, by default, on port 28443. End users are granted access by one of three means of authentication: session ID, flat-file and LDAP. The default authentication method is session ID. See “End User Authentication” in the *Manager Reference* for details on configuring an authentication method for the EUWI.

Administrators can control which features are available to users through the EUWI. For example, an administrator could choose to let users view a list of quarantined messages but not the contents of those messages.

Related tasks

[Configuring End User Authentication](#) (page 135)

1.5.9 Groups Web Interface

The PureMessage Groups Web Interface allows a global administrator to delegate administrative responsibilities to “group” administrators based on groups/domains and/or areas of responsibility. Delegated tasks can include quarantine management, reporting, list management and the configuration of certain policy settings.

The Groups Web Interface is a PureMessage service that runs, by default, on port 28443. Group Administrators can only access the tabs and features that have been made available by the global administrator.

1.5.10 Central Server Management

PureMessage can be deployed in multi-server configurations. “Server groups” consist of a central server and one or more [edge](#) servers. The edge servers are managed, configured, and maintained from the central server, which has connection and authentication profiles for the other machines. Centralized management of server groups is achieved through remote procedure calls (RPC) run from the central server.

By default, PureMessage has an “RPC User” account configured for the HTTPD (Manager) service. This user has a limited set of privileges, sufficient for starting and stopping services and updating configuration files.

Central server management makes it possible to:

- view the status of all PureMessage servers on the network
- add and remove edge servers
- start and stop PureMessage services on edge servers
- build and test configurations on the central server; distribute configurations to edge servers
- generate reports for edge servers (or groups of edge servers) from the central server

Publications

PureMessage uses “publications” to distribute the configuration files on the central server to edge servers. Edge servers are “subscribed” to these publications.

By default, PureMessage includes publications with configuration files for major aspects of the program’s functionality, such as Sophos-Anti-Virus-Conf and Anti-Spam-Config. Existing publications can be viewed and edited on the **Server Groups** tab of the PureMessage Manager.

Optionally, create custom publications using either the PureMessage Manager, or the `pmx-share` command-line program.

Related information

[pmx-share](#)

1.5.11 Reports

PureMessage offers pre-defined reports that provide graphical or tabular data on key performance statistics. For example, you can produce reports on “Top Spam Senders”, “Quarantine Size” and

“Memory Usage”. Reporting features are only available if the PostgreSQL database option was selected during installation.

Several PureMessage programs are used to populate certain database tables prior to initial use. Others consume report data from PureMessage log files and consolidate the data in the PostgreSQL database. There is also a program that automatically generates reports at scheduled intervals and emails them to the indicated recipients as specified in a scheduled job. The Installer or the [Scheduler](#) handles the configuration and compilation of most reports, but reports programs can also be run manually from the [command line](#) if necessary.

The **Reports** tab of the PureMessage Manager displays statistics for the currently selected report. There are drop-down lists for specifying the format and time-frame of a report.

For more about reports, see the “Reports” section of the *Manager Reference*.

Related concepts

[Reports Tab](#) (page 148)

1.5.12 Application Program Interface

PureMessage includes the `PerlMx` module, an interface for writing sendmail filter modules in Perl. It provides all of the “glue” code necessary to register a new filter with the sendmail Milter library and run the filter as a stand-alone process.

For a complete description of the callbacks and methods associated with the filter module, see the “PerlMx Development” appendix.

1.6 Contacting Sophos

Sophos Support

If you encounter a problem with your Sophos product, or if it does not function as described in the documentation, contact Sophos Technical Support: <http://www.sophos.com/support/>

Support for PureMessage Third-Party Applications

Some Sophos products ship with, or provide connections to, certain third-party software. Sophos Technical Support will respond to and work to resolve customer submitted issues that relate to the third-party applications: Sendmail, Postfix, JSMS and PostgreSQL in the manner described below.

Sendmail and Postfix Support

Sophos Technical Support will support only the Sophos-certified versions of sendmail or Postfix that are shipped with PureMessage. This support is limited to:

1. Providing help and guidance on the default configuration files, utilities, deployment scenarios, and user-developed configurations created using the PureMessage for UNIX graphical user interface or command-line tools.
2. Providing help and guidance on the best practices published in the PureMessage documentation.
3. Providing help and guidance on published upgrades and patches to Sophos-certified versions of sendmail and/or Postfix.

External/third party versions of Postfix are not quality-assured for integration with PureMessage, Sophos reserves the right not to provide support for an issue that appears to be related to any such custom configuration, and may recommend that you install the version of Postfix that is bundled with PureMessage to further a resolution.

PostgreSQL support

Sophos Technical Support will support only the Sophos-certified version of PostgreSQL that is shipped with PureMessage. Such support is limited to:

1. Providing help and guidance on the default configuration.
2. Providing help and guidance on the best practices published in the PureMessage official documentation.
3. Providing help and guidance on published upgrades and patches to the Sophos-certified version of PostgreSQL.

Other versions of PostgreSQL are not supported by Sophos Technical Support.

PureMessage Feedback

Please send comments and suggestions to puremessage-feedback@sophos.com

Help us in our continuous efforts to improve the accuracy of our spam heuristics by forwarding misidentified items *as an attachment* to:

- Missed Spam: is-spam@labs.sophos.com
- Not Spam: not-spam@labs.sophos.com

Corporate Contact Information

To contact your local Sophos office, see: <http://sophos.com/companyinfo/contacting/>

2 Managing PureMessage

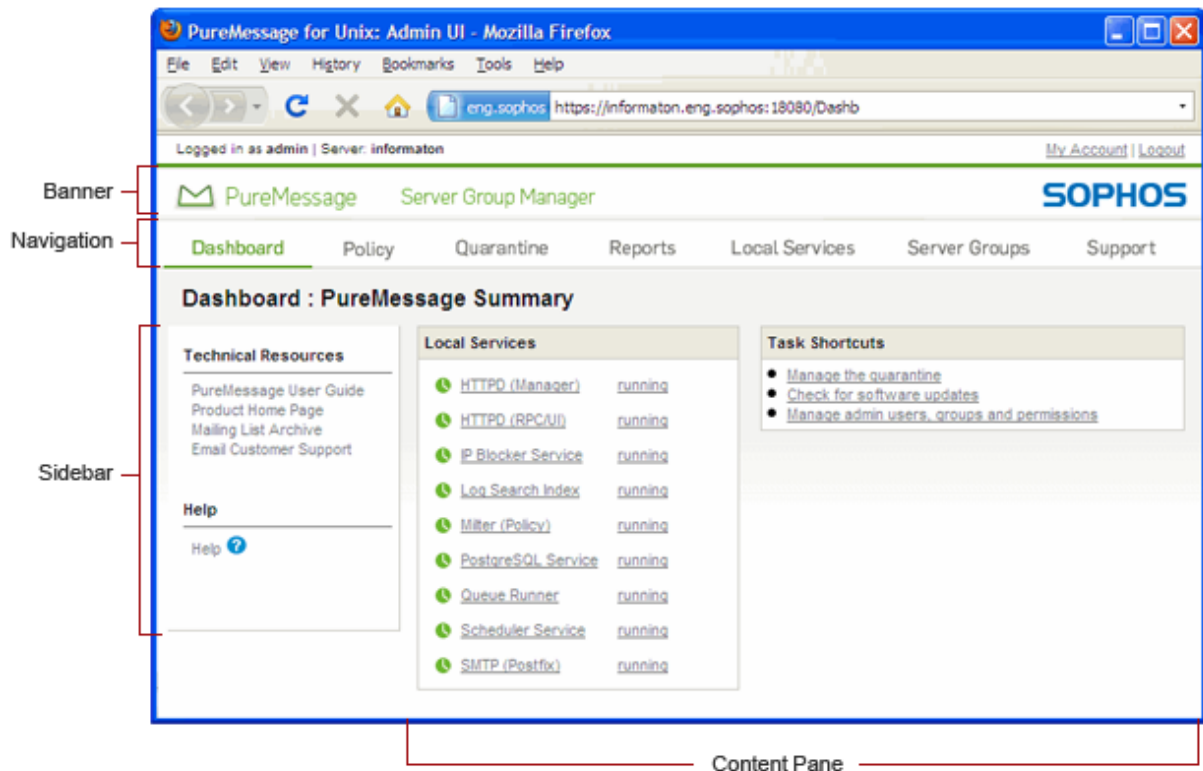
The PureMessage Manager is a web-based graphical interface to PureMessage. Although there are some administrative tasks that can only be performed from the command line, the majority of tasks can be accomplished via the Manager.

2.1 Dashboard Tab

The **Dashboard** tab displays a high-level overview of the PureMessage system that includes:

- **Local Services:** Displays a summary of the status of PureMessage services.
- **Task Shortcuts:** Provides quick links to commonly used features of the PureMessage Manager.
- **Technical Resources:** Provides links to PureMessage information and support resources.

2.1.1 Using the Manager Interface



The image above shows the PureMessage Manager, as viewed in a web browser. Note the labels for the basic parts of the Manager's graphical interface. They are referred to throughout the documentation and are as follows:

The Banner is at the top of the Manager. On the right side of the Banner are the **Logout** link and the **My Account** link. The latter lets you access a page where you can change your password (see "Changing Your Password" for more information).

The Sidebar on the left edge of every tab provides links to pages that can be displayed within the current tab, as well as links to help topics for the current page and links to other relevant parts of the PureMessage documentation.

The Content Pane displays the various pages of the PureMessage Manager interface, each of which contains information, configurable settings or management controls for specific aspects of PureMessage. Clicking a tab causes the default page for that tab to be displayed in the content pane.

The PureMessage Manager tabs and their functions are as follows:

- Dashboard tab provides a high-level overview of the status of the PureMessage system.

- Policy tab provides the ability to configure the PureMessage system's policy, which determines mail-filtering behavior.
- Quarantine tab provides capabilities to manage PureMessage quarantines, which are directories for the temporary storage of problematic (or potentially problematic) messages. Typically, quarantines are used for potential spam messages that can be reviewed later by administrators or end users to determine if they want to release (deliver) or delete them, or for virus-infected messages that you choose to retain for legal, organizational policy, or forensic reasons.
- Reports tab displays three of the key reports by default. It allows you to generate a wide variety of reports, export the report data for use in other applications, and schedule automatic report emails.
- Local Services tab provides a means to manage and configure the network services and scheduled jobs running on the local system.
- Server Groups tab provides capabilities to manage services on other hosts in a multi-server PureMessage system, as well as to publish policy and related configuration settings from the Central Server Manager (CSM) system to other PureMessage systems.
- Support tab provides a variety of PureMessage support and system maintenance information and capabilities.

2.1.2 Changing Your Password

To change the PureMessage Manager password from within the Manager interface:

1. Click My Account in the top right corner of the Manager banner.

The My Account page is displayed.

2. Type the new password into the Password and Confirm Password fields, and click Update.

A message appears above the Account Details form informing you that the password has been updated.

Note

The password must be a maximum of eight characters.

2.2 Policy Tab

The **Policy** tab allows you to configure the PureMessage system's message-filtering behavior, or "policy" using the Policy Constructor, a graphical editing tool. Alternatively, you can directly edit the policy script, located at `/opt/pmx6/etc/policy.siv`). Knowledge of the [Sieve](#) scripting language is required.

The policy is a structured series of rules, which are made up of tests and actions that are executed in the specified order. Rules can be added, modified, moved, or deleted. There are also tools for backing up the policy and lists before making changes, for testing policy changes, and for restoring previous policy configurations.

The policy makes use of various lists, which are used in the policy to exclude or include the specified hosts, users, or attachment types, or to scan for the listed offensive words. Maps are used to associate one email address with another for the purpose of redirection, or to selectively apply user preferences. Lists and maps can be created, modified or deleted.

In addition to editing the policy and lists, there are several pages of options and rules that control how spam and viruses are handled by PureMessage.

The **Policy Repository** page provides access to a collection of policy snippets that can be added to the PureMessage policy. You can create, modify and delete snippets.

2.2.1 Creating and Restoring Policy Backups

Before you edit the policy, or after you have successfully tested a policy change, you should back up the policy. If necessary, you can restore a backed up version of the policy.

To create a new policy backup:

On the sidebar of the **Rules** page (the default page of the **Policy** tab), click **Create** beside **Backups**.

An information message is displayed indicating that the backup was created, and a new sidebar entry appears in the **Backups** section named with the current date and time.

Note

Only the five most recent backups are kept. If you create a sixth, the oldest backup is automatically deleted.

To restore a policy backup:

1. On the sidebar of the **Rules** page (the default page of the **Policy** tab), click the policy backup that you want to restore.

An information message appears, asking if you want to restore the selected policy backup. You can click **Diff** to see a summary of the differences between the selected backup and the current policy to confirm that it is the backup that you want to restore.

2. Click **OK**.

You are returned to the **Rules** page, where a message displayed at the top of the page stating that "The backup has been successfully restored."

3. Click **Restart now** at the top of the page to restart the **Milter (Policy)** service.

2.2.2 Adding Policy Resources to Publications

Although most policy resources can be distributed to various hosts in a multi-server deployment, many of these resources must first be manually added to a PureMessage publication. When you are working with a particular policy resource (the policy, lists, maps anti-virus options, or anti-spam rules) a message appears near the top of the page stating one of the following:

- **This configuration is shared: n subscriber(s)**: These resources are currently included in a publication that can be shared with other PureMessage servers. Click "shared" to open the **Server Groups > Edit Publication** page for the publication that includes this resource.
- **You can publish this configuration to other hosts**. These resources are not currently included in a publication, but they can be added to one.

To add a policy resource to a publication:

1. In the top information banner, which states that **You can publish this configuration to other hosts**, select the publication to which you want to add the resource or select **Create New** from the drop-down list.
2. Click **Share**.

The **Server Groups: Edit Publication** page is displayed with the policy resource listed as a **Published Object**. If you chose an existing publication, the information for that publication is

displayed. If you chose **Create New**, the **Publication Name** text box contains an auto-generated name, and the **Description** is empty.

3. Enter or modify the **Publication Name** and **Description** and click **Save**.

The publication is added to the list of publications on the **Server Groups** sidebar.

Related concepts

[Managing Publications](#) (page 184)

2.2.3 Editing the Policy

Messages processed by PureMessage are passed through the policy filter. The policy filter compares characteristics of the message against policy tests, and, depending on the outcome of the test, applies the associated action. Together, tests and actions are referred to as "rules".

Note

Before editing the policy, make a backup of it, as described in "Managing Policy Backups".

Related tasks

[Creating and Restoring Policy Backups](#) (page 79)

Policy Rules: Order of Processing

Rules are processed in the order that they are displayed on the **Policy** tab. Unless a "Stop processing" action is configured, each rule will be tested regardless of the tests and actions of previous rules.

Given that rules are processed consecutively, the efficiency of the policy filter depends on the order of the rules, and the point at which rule processing stops. For example, if the first rule in the policy rejects messages that originate from a blacklisted host, it is not necessary (and is, in fact, inefficient) to continue processing the message against subsequent rules.

To change the order of existing rules within the policy: Click **Cut** in the original rule and then click **Paste** to reinsert it in the desired position.

Note

This version of the PureMessage Manager's Policy Constructor does not correctly display the position of actions when they are added to rules containing sub-rules. When a new action is added to a rule that has sub-rules, the action is displayed before the sub-rules in the Policy Constructor, but the action is actually added after the sub-rules in the `policy.siv` source file.

For example, in the default PureMessage policy, if an action were added to the "Mail from internal hosts" rule, the action would appear, in the Manager, within the main rule definition. However, if you view the `policy.siv` source, the new action is actually located after the sub-rule.

To fix the position of an action:

1. On the Policy Constructor page, click **see the source**.
2. Click the **filename** link at the top of the page to edit the `policy.siv`.
3. Manually position the new action to the desired place in the policy, and click **Save**.

About Tests

The test component of a policy rule specifies the *message characteristic* that should be analyzed, the *test expression* that is compared to that message characteristic, and the *operator* that defines the condition under which the test is true.

For example, a test might look for the word "Internal" in the message subject. In that case, the message characteristic would be "Subject," the test expression would be "Internal", and the operator would be "Contains".

Text boxes for specifying the operator and test expression are not displayed until the message characteristic is selected.

Multiple tests can be specified within a single rule. When multiple tests are configured, you are prompted to specify the cumulative result of all the tests. Choose from one of the four options in the drop-down list:

- If ALL criteria are met: All of the tests must be true for the rule's action(s) to be performed.
- If ANY criteria are met: Only one of the tests must be true for the rule's action(s) to be performed.
- If NO criteria are met: None of the tests can be true for the action(s) to be performed.
- Not ALL criteria are met: At least one of the tests must not be true for the action(s) to be performed.

Whenever multiple tests are specified, PureMessage displays a set of buttons beneath each test. Use the up and down arrow buttons to change the order in which the tests are performed. Use the "X" button to delete a specific test.

Note

As of PureMessage 5.2.1, some tests allow you to specify Japanese characters. See the version 5.2.1 section of the Release History for details.

Related concepts

[About Actions](#) (page 93)

[Message Characteristics](#)

'To' Address

Analyze the To headers in the message. To analyze all message recipients, including Cc and Bcc recipients, use the Recipient's address test.

(Supports Multiple Test Expressions. For more information, see "Test Expressions")

Always Match

Do not analyze the message. This test is always true. The action is performed regardless of the message characteristics.

Attachment name

Analyze the Content-Type and Content-Disposition headers and, when using true file type identification, the contents of the files themselves. This test determines the filename of each message attachment. Expands the %%ATTACHMENT_NAMES%% template variable.

The **Arguments** button exposes the following options:

- Use true file type identification for filename extensions: Modifies the test to use true file type detection for filename extensions. The files themselves are examined, as well as the filenames shown in the Content-Type and Content-Disposition headers. Archives are automatically expanded, so that files within other files can be tested. PureMessage will search as many levels as necessary (up to a configured maximum) to find specified file types. The maximum recursion depth is set in `/opt/pmx6/etc/tft.conf`.

Tests for document types should use the application-specific file extension (such as `.doc` or `.xls`). To view a list of extensions that PureMessage supports, run:

```
pmx-list-true-filetypes --verbose
```

- Match-type: The match operator. For a description of options contained in the drop-down list, See the "Operators" section.
- Match values: The file type(s) to match.

Note

Because this test also returns true whenever PureMessage is unable to scan a message, you should use the "Message failed to scan" test within the "Attachment name" test to specify how unscannable messages are handled.

(Supports Multiple Test Expressions. For more information, see "Test Expressions")

Attachment size

Analyze the size of each message attachment. This test applies to any MIME attachment, including text/plain and text/html message body parts. Expands the %%ATTACHMENT_NAMES%% template variable.

Attachment true filetype

Scan the content of message attachments, including archived attachments such as `.zip` and `.tgz` files. Archives are automatically expanded so that files within other files can be tested. PureMessage searches as many levels as necessary (up to a configured maximum) to find specified file types. The maximum recursion depth is set in `opt/pmx/etc/tft.conf`. Expands the %%ATTACHMENT_NAMES%% template variable.

The advantage of using this test instead of similar tests (Attachment type and Attachment name) is that the action is not performed on a message unless the file type is a true match for one that is specified in this test. For example, the Attachment type test will perform the action if there is a file type match in the message's Content-Type header, even if that does not represent the true identity of the file. So, it could be that a message appears to contain a Microsoft Word attachment, but the file extension has been falsified, and the message actually contains a `.jpeg` file instead.

This test can be used to detect specific file types or groups of file types. To view a list of supported file groupings, run:

```
pmx-list-true-filetypes
```

To view the specific file extensions within those groupings, run:

```
pmx-list-true-filetypes --verbose
```

Important

When using this test, you should specify the appropriate true filetype definition (as displayed in the list of groupings) for the match-type. For example, if you wanted to perform an action on all variations of Microsoft Word files ranging from Word 95 to the most recent Word document extensions, you could specify **Contains** as the match-type and `Document/Microsoft Word` as the match value. Or, to create a rule that applies the same actions to any message with an attached image, you could specify **Contains** as the match-type and `Image/` as the match value.

The **Arguments** button exposes the following options:

- Match-type: The match operator. For a description of options contained in the drop-down list, See the "Operators" section.
- Match values: The file type(s) to match.

Note

Because this test also returns true whenever PureMessage is unable to scan a message, you should use the "Message failed to scan" test within the "Attachment true filetype" test to specify how unscannable messages are handled.

Attachment type

Analyze the Content-Type and Content-Disposition headers and, when using true file type identification, the contents of the files themselves. This test determines the type of each message attachment. Expands the `%%ATTACHMENT_NAMES%%` template variable.

The **Arguments** button exposes the following options:

- Use true file type identification for file type: Modifies the test to use true file type detection. The file itself is examined in addition to the declared Content-Type and Content-Disposition headers. Archives are automatically expanded, so that files within other files can be tested. PureMessage searches as many levels as necessary (up to a configured maximum) to find specified file types. The maximum recursion depth is set in `/opt/pmx6/etc/tft.conf`.

Tests for attachment types should use the Mime file type. For a complete list of the Mime types that PureMessage supports, run:

```
pmx-list-true-filetypes --verbose
```

- Match-type: The match operator. For a description of options contained in the drop-down list, See the "Operators" section.
- Match values: The string or string list of file types to match.

Note

Because this test also returns true whenever PureMessage is unable to scan a message, you should use the "Message failed to scan" test within the "Attachment type" test to specify how unscannable messages are handled.

(Supports Multiple Test Expressions. For more information, see "Test Expressions")

Body size

Analyze the size of the body of the message.

Content-Type

Analyze the value of Content-type headers in the message.

Envelope from

Analyze the Envelope From value in the message.

(Supports Multiple Test Expressions. For more information, see "Test Expressions")

Envelope group

Analyze the Sender or Recipient(s) Group(s), depending on the message's direction, and match them against the specified group.

(Supports Multiple Test Expressions. For more information, see "Test Expressions")

Envelope to

Analyze the Envelope To value in the message. Specify individual recipients or lists of recipients. However, if a message addressed to a number of recipients tests true, specified actions are performed for all recipients. If, for example, a message with an attachment is addressed to five recipients, two of which match a list specified in the "Envelope to" test, and an "Drop attachment" action is also specified, none of the five recipients receive the attachment.

Important

When using this test in conjunction with email addresses that have been associated as part of a Recipient aliases map, you must ensure that the Envelope To value specified here matches the "Map To" portion of the address map. For example, if `service@example.com` has been mapped to `joe@example.com`, then the Envelope To address is `joe@example.com`. For more information, see "Address Maps" in the Policy Rules section of the *Administrator's Reference*.

(Supports Multiple Test Expressions. For more information, see "Test Expressions")

Header contains word or phrase

Check the header for a specified word or phrase.

The **Arguments** button exposes the following options:

- **Comparator:** If selected, you can specify that the match be case sensitive. To specify case sensitivity, you must also enter `i;octet`. If the accompanying check box is not selected or it is empty, the match is case insensitive.
- **Match-type:** The match operator. For a description of options contained in the drop-down list, See the "Operators" section.
- **Header names:** The specific email headers in which to search.
- **Match values:** The string or string list of file types to match.

(Supports Multiple Test Expressions. For more information, see "Test Expressions")

Header exists

Check for the occurrence of a specific header.

(Supports Multiple Test Expressions. For more information, see "Test Expressions")

Header size

Analyze the size of the message header.

(Supports Multiple Test Expressions. For more information, see "Test Expressions")

Message contains a virus

Scan the message for any viruses.

Message contains credit card number

Scan the message for numeric sequences that are common to major credit cards. This test is based on the Luhn algorithm, a checksum formula that performs the validation. Supported credit cards are American Express, MasterCard, Visa, Visa Electron, Discover Card, Diners Club, and China Union Pay.

The `creditcard_number_limit` setting in `/opt/pmx6/etc/creditcard.conf` determines how many numeric sections PureMessage will scan in a given message. By default, PureMessage scans nine separate sections of numbers that might contain credit card numbers before proceeding to the next stage of processing.

You can configure the scan failure actions for this test in `/opt/pmx6/etc/scanlimit.d/phrase.conf`.

The **Arguments** button exposes the following options:

- **Search attachments:** When selected, this test also matches inside of attachments, including attachments contained in `.tgz` and `.zip` files. For example, it can detect a phrase that appears in an `.xls` file, which is embedded in a `.doc` file that is inside a `.zip` file.

- Search all attachments, even after a match: When specified along with the Search attachments option, this test scans all attachments regardless of whether a match is found. If you only specify Search attachments, the associated action is performed as soon as the first match is found, so any remaining attachments are not scanned. This is a concern if you are combining the “Message contains word or phrase” test with actions such as “Drop attachment” because only the attachment in which the first match occurred would be dropped; the remainder would be delivered, regardless of their contents.

Note

Because this test also returns true whenever PureMessage is unable to scan a message, you should use the “Message failed to scan” test within the “Message contains credit card number” test to specify how unscannable messages are handled.

Message contains suspicious attachments

Check message attachments for filenames or file extensions specified in the **Suspect Attachment Names** list and check Content-Type and Content-Disposition headers for attachment types specified in the **Suspect Attachment Types** list.

The **Arguments** button exposes the following options:

- Use true file type identification: The files themselves are examined in addition to the declared Content-Type and Content-Disposition headers. Archives are automatically expanded, so that files within other files can be tested. PureMessage will search as many levels as necessary (up to a configured maximum) to find specified file types. The maximum recursion depth is set in `/opt/pmx6/etc/tft.conf`.

Most common archive formats are supported. If you are creating a new rule that tests for suspect attachments, it is recommended that you specify this argument instead of the deprecated “Inspect archives” argument described below.
- Inspect archives: Enables this test in archive files (such as .zip files) attached to messages, even if true file type identification is not enabled. The test attempts to inspect the contents of an attached archive file for files that match the suspect-attachment-names list. Only .zip archives (plain and encrypted) are supported, and only the top-level archive is inspected. If the archive contains a nested archive, the nested archive is not inspected.

Note

Because this test also returns true whenever PureMessage is unable to scan a message, you should use the “Message failed to scan” test within the “Message contains suspicious attachments” test to specify how unscannable messages are handled.

Message contains the specified virus

Analyze the stated virus names; automatically runs the “Message contains a virus” test.

(Supports Multiple Test Expressions. For more information, see “Test Expressions”)

Message contains unscannable data

Returns true if virus scanning for a message fails, and no viruses were found in the message. This test only works if it is preceded by the test "Message contains a virus." It is used to differentiate between messages that cannot be scanned for some reason (for example, encryption), and messages that contain viruses (either instance causes the test to return true).

Specify the types of unscannable content that PureMessage will allow or deny by editing the `cantscan.conf` file.

Message contains word or phrase

Only the "contains" and "Matches regex" Operators are recommended for this test. The "is" and "matches" tests compare against the entire text of the message, which is usually not desirable when looking for a particular phrase.

The **Arguments** button exposes the following options:

- **Search attachments:** If selected, this test also matches inside of attachments, including attachments contained in `.tgz` and `.zip` files. For example, it can detect a phrase that appears in an `.xls` file, which is embedded in a `.doc` file that is inside a `.zip` file.
- **Search all attachments, even after a match:** When specified along with the Search attachments option, this test scans all attachments regardless of whether a match is found. If you only specify Search attachments, the associated action is performed as soon as the first match is found, so any remaining attachments are not scanned. This is a concern if you are combining the "Message contains word or phrase" test with actions such as "Drop attachment" because only the attachment in which the first match occurred would be dropped; the remainder would be delivered, regardless of their contents.
- **Match-type:** The match operator. For a description of options contained in the drop-down list, See the "Operators" section.
- **Match values:** The string or string list of file types to match.

You can set the maximum attachment size and the maximum time per message scanned in `/opt/pmx6/etc/phrase.conf`.

Note

Because this test also returns true whenever PureMessage is unable to scan a message, you should use the "Message failed to scan" test within the "Message contains word or phrase" test to specify how unscannable messages are handled.

(Supports Multiple Test Expressions. For more information, see "Test Expressions")

Message failed to scan

Returns true if a message could not be scanned. This test is only available for use within the following content tests: Attachment name, Attachment true filetype, Attachment type, Message contains credit card number, Message contains word or phrase, or Message contains suspicious attachments.

This test is mainly used to differentiate between messages that cannot be scanned for some reason (e.g., an unrecognized file type), and messages that contain undesired content, since both will cause the various content tests to return true. The kinds of unscannable content that PureMessage should

allow or deny can be specified by editing the files in `/opt/pmx6/etc/scanlimit.d/`. To configure unscannable content options for the Message contains word or phrase and Message contains credit card number tests, use `/opt/pmx6/etc/scanlimit.d/phrase.conf`. For the attachment tests, use `/opt/pmx6/etc/scanlimit.d/tft.conf`.

Message has offensive content

Analyze the visible text in a message; compare it to the contents of the **Offensive Words** List. This test decodes base64/quoted-printable encoded text and strips out HTML markup before looking for a match.

Message is from blocked IP

Checks the sender's IP address against IP blocklist data from Sophos Labs. IP addresses defined in the **IP Blocking Exception**, **Trusted Relay IPs** and **Internal Hosts** lists are exempted.

This test is a policy-level alternative to MTA-level IP blocking. It is only effective if the IP Blocker Service is running. Using this test in the policy allows more flexibility in handling messages from blocked IP addresses, but it is not as efficient as rejecting the messages at the MTA level.

Even if you choose to block IP addresses at the policy level, it is still recommended that you enable the expanded IP-blocking functionality described in "Enabling or Disabling MTA IP Blocking" in the Local Services Tab section of the *Manager Reference*.

Message size

Analyze the total message size.

Never match

Do not analyze the message. This test is always false. The action is performed regardless of the message characteristics.

Number of attachments

Analyze the total number of message attachments.

Number of recipients

Analyze the total number of message recipients.

Preamble length

Analyze the number of characters in the preamble area of a multipart message.

Epilogue length

Analyze the number of characters in the epilogue area of a multipart message.

MIME specification

Analyze the message content. Use to check that the message is in MIME format.

Percentage of 8-bit characters

Analyze the total number of 8 bit (non-ASCII) characters in the message body. Use to check whether a message is 7 bit-clean (pure ASCII).

Received header

Analyze the Received headers in the message.

Recipient's address

Analyze the To, Cc and Bcc headers in the message.

(Supports Multiple Test Expressions. For more information, see "Test Expressions")

Relay

Analyze the hostname or IP address of the server that passed the message to the local domain.

(Supports Multiple Test Expressions. For more information, see "Test Expressions")

Reply-to header

Analyze the Reply-to headers in the message.

(Supports Multiple Test Expressions. For more information, see "Test Expressions")

Reinject a delayed message

Reinjects the message that was processed and delayed earlier by PureMessage militer service.

Sender's address

Analyze the From headers in the message.

(Supports Multiple Test Expressions. For more information, see "Test Expressions")

Spam probability

Calculate the message's spam probability. If a message passes through several Spam probability tests, the message is only scanned once for its spam probability; its score is saved. This makes it possible to have different actions based on different spam probability ranges without having to scan the message multiple times.

Spam rule hit

Analyze the names of spam rules violated by the message; automatically performs the Spam probability test. Refer to the **Anti-Spam Rules** page for a list of configured rules.

(Supports Multiple Test Expressions. For more information, see "Test Expressions")

Subject

Analyze the contents of the message subject.

(Supports Multiple Test Expressions. For more information, see "Test Expressions")

Verify Message DKIM Header

Important

To ensure the effectiveness of this test, it is recommended that you insert it ahead of spam tests in the policy. It is particularly crucial that this test be run before tests that modify message headers because such modifications could corrupt or remove the DKIM signature.

DomainKeys Identified Mail (DKIM) is an authentication framework used to sign and validate a message, based on the domain of the sender. This test performs the validation by verifying the origin of the sending address. To configure PureMessage to add a DKIM signature to some or all outgoing messages, see **Sign message with DKIM header** in the "Actions Defined" section. DKIM verification depends on access to public keys stored on an available DNS server. This must also be configured for the test to take affect. For general information, see www.dkim.org.

The test analyzes the message header for DKIM signatures. It allows PureMessage to verify the origin of messages that bear a DKIM signature. This test returns true if the message matches the value specified.

Create multiple tests to allow for the various possibilities described in the values below.

Configurable Values:

When using this test with match operators such as **Contains** and **Matches**, type one of the following four values in the adjacent text box.

- pass - A signature is detected and verified.
- none - there is no DKIM signature.
- fail - A DKIM signature is present but cannot be verified.
- invalid - The message cannot be verified for some reason. For example, a DNS timeout or an invalid DNS hostname.

If you plan to quarantine messages based on certain test results, you may want to create a quarantine digest informing end users that messages have been quarantined for this reason. For more information, see “Managing Quarantine Digest Rules” in the Quarantine Tab section of the *Manager Reference*.

Related concepts

[Test Expressions](#) (page 91)

[Operators](#) (page 91)

[Template Variables](#) (page 101)

[Address Maps](#) (page 253)

[Actions Defined](#) (page 93)

[Managing Quarantine Digest Rules](#) (page 142)

Related tasks

[Enabling or Disabling MTA IP Blocking](#) (page 178)

Related information

www.dkim.org

[pmx-policy](#)

[cantscan.conf](#)

[creditcard.conf](#)

[phrase.conf](#)

[pmx.conf](#)

[tft.conf](#)

Test Expressions

The message characteristic and operator settings determine the type of data that is entered as the test expression. For example, if the message characteristic being tested is the **Attachment name**, and the operator is **Matches regex**, enter a regular expression as the test expression. If the message characteristic being tested is the **Spam probability** and the operator is **Is over**, enter a number between 1 and 99 as the test expression.

Multiple Test Expressions

Some tests allow multiple test expressions. (If a test supports this, it is noted in the message characteristic description.) To specify multiple test expressions, enclose each expression in quotation marks, and separate them with commas. For example, to compare the selected message characteristic against two email addresses, enter the addresses in the following format:

```
"address@domain.com","address2@domain.com"
```

Operators

The match operator determines the type of comparison that is made between the message characteristic and the test expression. The match operator determines the nature of the data that should be entered as the test expression.

The available match operators are either strings and lists or numerical expressions, depending on the nature of the message characteristic being tested. For example, if the test analyzes the size of a message component, the operator list contains numerical expressions. If the test analyzes the text of a message, the operator list contains string and list expressions.

Depending on the selected operator, different types of text boxes are displayed for entering test expressions. For example, selecting the **Is a member** of match operator causes the display of a drop-down list from which you can select a policy test expression list. Selecting **Matches regex** reveals a

text box where a regular expression is entered as the test expression. Selecting **Is under** reveals a text box where a number can be entered.

Match Operators - Strings and Lists

- **Contains:** Compare the selected message characteristic with the specified test expression; perform the action if the test expression occurs within the message characteristic.
- **Does not contain:** Compare the selected message characteristic with the specified test expression; perform the action if the test expression does not occur within the message characteristic.
- **Is:** Compare the selected message characteristic with the specified test expression; perform the action if the test expression is an exact match.
- **Is not:** Compare the selected message characteristic with the specified test expression; perform the action if the test expression is not an exact match.
- **Matches:** Match using wildcards. The "?" character matches against any single character; the "*" character matches any group of characters. The rule is true if the test expression matches the message characteristic.
- **Does not match:** Match using wildcards. The "?" character matches against any single character; the "*" character matches any group of characters. The rule is true if the test expression does not match the message characteristic.
- **Is a member of:** Compare the message characteristic to items in the specified list; perform the action if the message component is found in the list.
- **Is not a member of:** Compare the message characteristic to items in the specified list; perform the action if the message component is not found in the list.
- **Matches regex:** Compare the message characteristic to the specified regular expression; perform the action if it matches. For example, a regular expression can be used to test the contents of a message's Envelope to field. When regular expressions are used in policy rule tests or actions, they are not prefixed or suffixed with slashes or braces. However, if you are manually editing the policy script on the command line, you must "escape" backslashes and quotes within regular expressions by preceding them with a backslash. (The PureMessage Manager automatically escapes these characters.) For example:

— Correct:

```
if (header "foo" :re "bar") if (header "content-type" :re
                             "text\\./plain")
```

— Incorrect:

```
if (header "foo" :re "/bar/") if (header "content-type" :re
                             "text\\/plain")
```

See the Regular Expression Primer Overview for more information on using regular expressions within the PureMessage policy.

- **Does not match regex:** Compare the message characteristic to the specified Regular Expression; perform the action if it does not match. When regular expressions are used in policy rule tests or actions, they are not prefixed or suffixed with slashes or braces. You must escape backslashes and quotes with a backslash character. See Matches regex, above, for examples on using regular expressions within policy tests. See the "Regular Expression Primer" for more information on using regular expressions within the PureMessage policy.

Match Operators - Numbers

Numerical operators compare the numerical value of the specified message characteristic to the number entered in the expression text box. The numerical operators are greater than (>), greater than or equals to (>=), less than (<), and less than or equals to (<=).

When analyzing the size of message components, the default unit of measure is bytes. To analyze size based on a different unit of measure, append "K" (kilobytes), "M" (megabytes) or "G" (gigabytes) to the number entered in the test expression text box.

About Actions

Actions are executed if the test specified in a rule is true. Multiple actions can be defined within a single rule; if the test result is true, all of the actions are performed.

Whenever multiple actions have been specified, PureMessage displays a set of buttons beneath each action. Use the up and down arrow buttons to change the order in which the actions are performed. Use the "X" button to delete a specific action.

Related concepts

[About Tests](#) (page 81)

Actions Defined

The following actions are available from the **Execute actions and rules** drop-down lists available in the PureMessage Manager's Policy Constructor.

Note

As of PureMessage 5.2.1, some actions allow you to specify Japanese characters. See the version 5.2.1 section of the Release History for details.

Accept the message

Deliver the message to its envelope recipients.

Add banner

Add a custom banner to the body or header of a message. If the message contains both an HTML and a plain text part, the banner is added to both (unless Ignore these content-types is used). If the message contains multiple text parts, the banner will be added to the first one.

- Append banner to message body: Add the banner to the end of the message body. For HTML parts, the banner is inserted just before the </BODY> tag. If there is no </BODY> tag, the banner is inserted just before the </HTML> tag.
- Prepend banner to message body: Insert the banner at the beginning of the message body. For HTML parts, the banner is inserted immediately after the <BODY> tag. If there is no <BODY> tag, the banner is inserted immediately after the <HTML> tag.

Note

Choose either Append banner to message body or Prepend banner to message body, not both.

- **Banner character set:** Specify the character set of the banner so it is only added to matching message parts. This field can be a string or a regular expression that matches several aliases of a particular character set. The test is not case sensitive. For example, "Big[_-]5.*" would match and add a banner to message parts declared as "big5", "Big-5", and "BIG5-HKSCS". If a character set is not specified, the banner text is added to all message parts. If the specified character set does not match any message parts, then no banner is added.
- **Add banner to specified header:** Add the banner to a message header; specify the header name.
- **Re-code header containing encoded word(s):** If a header contains encoded words, re-encode it when adding the banner. This option should be used if adding the banner garbles the original encoded header when prepending text.
- **Ignore these content-types:** Ignore a content-type. For example, to add banners to text/plain parts only, specify "text/html".
- **Enclose banner in HTML <PRE> tags:** When adding the banner to an HTML part, put the banner in HTML <PRE> tags. This preserves text formatting (for example, newlines and indentation) that would otherwise be lost in the rendered HTML.
- **Data Type:** Specify whether the text entered in the File or String field is a string ("Verbatim"), or a filename ("Filename").
- **File or String:** If "Verbatim" was specified in the **Data Type** field, enter the text of the banner in this field (ASCII or Latin1 only). If **Filename** was specified in the **Data Type** field, specify the full path and filename. The contents of the file must be text. The file *must* be of the same charset specified in the policy action.

Note

If the banner is specified as verbatim and charset is used, it will only be added successfully if:

- iconv is enabled in utf8.conf
- charset is one of:
 - UTF-8
 - iso-2022-jp
 - euc-jp
 - Shift-JIS
 - iso-8859-1

Conversion from UTF-8 to iso-2022 does not work correctly for older versions of iconv. To work around this limitation, use an iso-2022-jp file for adding the banner.

Add header

Add a header to the message. Specify the name of the header, then the value the header should contain.

Use template variables to specify the value(s) to display in the message header.

Add recipient(s)

Adds a CC (carbon copy) recipient to the message, as opposed to "Redirect the message" (which replaces the envelope recipients) and "Forward to" (which adds a blind carbon copy recipient).

- **Recipient Address:** Specify the recipient address. To specify multiple recipients, enclose each recipient's address in quotation marks, and separate them with commas.

Archive message to a file (Extended Policy Model)

A copy of the message is stored in "mbox" format (compatible with many mail readers) in the specified directory and filename.

- **Original Message (No Modifications):** When selected, the original, unmodified message is archived. If this option is not specified, PureMessage includes all modifications made to the message up to that point in the policy script.
- **Archive the message to a 'message store' directory:** When selected, the message is saved using the PureMessage "message store" directory and file architecture. If a message store does not already exist, it is created in the location specified in the File Name text box (described below). For more about the 'message store', see the `pmx_archive` section of the `pmx-policy` man page.
- **File Name:** The path to the archive file. If the file already exists, it is overwritten. Any text that appears following the final forward slash (/) is interpreted as the filename. If an absolute path is not specified, the path is interpreted as relative to the home directory of the PureMessage user (`/opt/pmx6/home`). You must specify a path or filename.

Clean the message of any viruses

Tells the virus engine to clean the virus from the message. If cleaning fails, the message is quarantined, and a message is sent to the recipient based on the specified Failure Template File.

Clean the message of any viruses. All installed engines capable of cleaning will be used to attempt to clean viruses from the message. After cleaning, the message is rescanned to ensure that the clean was successful. If the virus engines are not able to clean the message, the part containing the virus is replaced by data from the specified Failure Template File.

- **Failure Template File:** Specify `cantclean.tmpl`, which is the default template failure file. If you create a custom template file, put it in the appropriate language-specific `/opt/pmx6/etc/templates/<language>/virus.d` directory or specify a full path.

Collect attachment statistics in message log

Writes an entry in the message log that includes:

- attachment name
- attachment type
- attachment size
- number of attachments

Copy the message to quarantine

Store a copy of the message in the quarantine. This command does not affect the delivery of the message. If called multiple times, multiple copies of the message are stored in the quarantine. The copy stored in the quarantine incorporates any changes made to the message as a result of actions that have occurred to that point.

- **Quarantine Reason:** Specify the reason for quarantine.

Custom policy mark

Mark the message with a key-value pair. This action should only be used for custom policy reports. For more information, see "Viewing and Managing Reports" in the Administrative Groups section of the *Administrator's Reference*. The message will accumulate marks as it is processed that are written to the message log. These marks can be used to generate custom statistical reports.

- **Log only one mark per message for the specified key:** When selected, PureMessage adds a single mark to the message log per message, for any given key. Regardless of the number of actions associated with the key, only one mark is logged.
- **Key:** Enter the key for the mark.
- **Value:** Enter the value to associate with the key.

Delete header

Delete the specified header.

- **Header Name:** Specify the name of the header.
- **Index:** If the header occurs multiple times in the message, specify which occurrence should be deleted. The first occurrence is "0", the second "1", and so on.

Deliver immediately for

Allows the message to be exempted from further processing for the specified recipients. The "envelope to" address is compared against the list or individual addresses specified in the Arguments dialog box. For example, to exempt a user's email from being checked for spam, create a Deliver immediately for action before the spam rule.

The message will be queued for delivery and delivered when the `pmx-queue` program is run (`pmx-queue` is configured as a scheduled job).

- **Except:** This reverses the exemption. That is, messages for all matches except those specified below will be exempted from further processing.
- **Match-type:** Only the "positive" match types (**Contains**, **Is**, **Is a member of**, **Matches** and **Matches regex**) can be specified. For example, the **Deliver immediately for** action supports the use of a regular expression as a match type for exceptions. When regular expressions are used in policy rule tests or actions, they are not prefixed or suffixed with slashes or braces. However, if you are manually editing the policy script on the command line, you must "escape" backslashes and quotes within regular expressions by preceding them with a backslash. (The PureMessage Manager automatically escapes these characters.) See the "Regular Expression Primer" for more information on using regular expressions within the PureMessage policy.
- **Key-list:** Enter a list ID or one or more email addresses in this field. To specify multiple addresses, enclose each address in quotation marks, and separate them with commas.

Discard the message

Tells the mail transfer agent to discard the message.

Drop attachment

Discard the attachment, but deliver the message. This action can only be used with tests that check message attachment characteristics. If the message has multiple attachments, the attachment associated with the test will be dropped.

Delay the message

Delays the message for the period specified by anti-spam engine. The message will be reinjected into the filter for policy processing after its delay period times out.

Forward to

Forward message (via a blind carbon copy) to the specified addresses. This action writes a copy of the message to the outgoing queue, which is then delivered.

- Address: Specify the address. To specify multiple recipients, enclose each address in quotation marks, and separate them with commas.

Log the message with key/value pair

Mark the message with a key-value pair. The message will accumulate "marks" as it is processed that are written to the message log when the message stops processing. These marks can be used to generate Policy Mark Hits reports, which show a count of keywords and keys from the message log.

- Key: Enter the key for the mark.
- Value: Enter the value to be attached to the key.

Log the message with keyword

Mark the message with given key. The message will accumulate "marks" as it is processed that are written to the message log when the message stops processing. These marks can be used to generate Policy Mark Hits reports, which show a count of keywords and keys from the message log. The keyword string can consist of alphanumeric characters and underscores, up to a maximum of 64 characters.

- Key: Enter the key value.

Map recipients

Maps the envelope recipients against the specified address map. The envelope recipients are looked up in the address map; if they match a source address, they are replaced with the destination address defined in the address map.

Note that address maps can be configured with an empty Map To value, which has the result of deleting the message.

- Map ID: Specify the ID code for the desired map.

Notify

Send a notification email to the sender or recipients of a message. The following fields can be configured for notifications:

- Notify who?: Specify whether to send the notification to the recipients or the sender of the message.
- Data Type: Specify whether the text entered in the Notification data field is a string ("Verbatim"), or a filename ("Filename").
- Map: To map the addresses specified in the Notify who? field against an address map, check the box and enter the desired map ID.
- Notification Data: If "Verbatim" was specified in the Data Type field, enter the text of the notification in this field. If "Filename" was specified in the Data Type field, specify the full path and filename. The contents of the file must be plain text.

Quarantine the message

Copy the message to the quarantine; do not deliver the message to the intended recipient(s).

- Reason: Enter the reason for the quarantine.

Redirect the message

Replaces all the envelope recipient addresses with the specified address.

- Address: Enter the desired address. To specify multiple addresses, enclose each address in quotation marks, and separate them with commas.

Reject the message

Tells the mail transfer agent to reject the message.

- SMTP Return Code: Optionally, enter a SMTP return code parameter for MTA response. The default is 550.
- ESMTP Status / Error Code: Optionally, enter an Extended SMTP status/error code for MTA response.
- Reason: Specify a reason for the rejection. Senders then receive a reason when a message 'bounces' back to them.

Rename attachment

Rename an attachment. The replacement string can use the %%ATTACHMENT_NAME%% template variable to provide the original filename in the replacement name (for example, %%ATTACHMENT_NAME%%.warning)

Replace header

Replace the value of the specified header with the specified value. See the Add header action for a list of template variables that can be used for the value. If the header does not exist, this action will add it.

- **Index:** If the header occurs multiple times in the message, specify which occurrence should be deleted. The first occurrence is "0", the second "1", and so on.
- **Header Name:** Specify the header name.
- **Header Value:** Specify the new value for the header. If no value is specified, the header will not be removed; it will simply have no value.

Replace message part

Replace an attachment, or the whole body of the message, with custom text or a file. If used in the same rule as an attachment test, the matching attachment(s) will be replaced. If it is used outside of this context, the entire body of the message will be replaced.

- **New Body Content-Type:** Specify the content-type to use in the replaced part. The default is "text/plain".
- **New Body Transfer-Encoding:** Specify the transfer encoding to use in the replaced part. The default is "7bit". Supported transfer encodings: UTF-8, 7bit, 8bit, base64, quoted-printable, binary.
- **Data Type:** Specify whether the text entered in the File or String field is a string ("Verbatim"), or a filename ("Filename").
- **File or String:** If "Verbatim" was specified in the Data Type field, enter the text of the notification in this field. If "Filename" was specified in the Data Type field, specify the full path and filename. The contents of the file must match the specified body content type.

The following template variables can be used in the file or string.

- **%%ATTACHMENT_NAME%%:** The filename of the attachment (as given by the Content-Disposition header)
- **%%ATTACHMENT_TYPE%%:** The content-type of the attachment (as given by the Content-Type header)
- **%%ATTACHMENT_SIZE%%:** The size of the attachment

Route message

Route the message to a specified server or to multiple servers, usually for the purpose of archiving or encrypting. Enter the IP address(es) or fully qualified hostname(s) of the server(s) to which the message should be routed. For example, you may want make a copy of each message sent to the customer service department and archive it on a separate server.

- **Server(s):** Enter the IP address(es) or fully qualified hostname(s) of the "route to" server(s).
- **Route a copy of the message:** In addition to processing the message and sending it to its intended destination, a copy of the message is also sent to the specified server(s). If this option is not used, the message will be routed to the specified server instead of the intended recipient(s).
- **Disable bounces:** If the specified routing server is not available, do not bounce the message. By default, the message is bounced to the original sender. The original sender receives a notification that is defined in the template `/opt/pmx6/etc/templates/Language_Dir/bounce-on-failure.tmpl`.

Important

When using the “Route message” action, make sure that the relay you specify here is not also a mail server that is delivering mail via the same server that performed the routing action. This would cause PureMessage to route mail in a continuous loop instead of delivering it.

Note

If, when upgrading to the latest version of PureMessage, an error message is displayed regarding the `smtp_generic_maps` setting in `/opt/pmx6/postfix/etc/main.cf`, complete the following steps:

1. Because policy routing only works with "pcre" maps, ensure that the `smtp_generic_maps` setting is of the type `pcre`. For example:

```
smtp_generic_maps=pcre:PathToMapFile
```

2. In the pcre map file, add the following regular expression on a separate line at the end of the file:

```
/ (.* )% (.* )@.*/ $1@$2
```

3. As the root user, in `/opt/pmx6/postfix/etc`, run `make`.

These steps may also be necessary if you are using an external Postfix installation. Follow the same instructions, making adjustments for the different file locations.

Set a template variable

Initialize a template variable with a shell command. This action will execute the shell command and put the output in a template variable that can be used in subsequent tests and actions. Use the `%MESSAGE_FILE%` template in the shell command to determine the filename of a file containing a message.

- **Variable:** Specify the template variable name. Can only consist of upper-case alphanumeric characters and underscores.
- **Shell Command:** Specify the shell command to run.

Sign message with DKIM header

DomainKeys Identified Mail (DKIM) is an authentication framework used to sign and validate a message, based on the domain of its sender. This action signs outgoing messages with a unique DKIM signature. To configure PureMessage to detect and verify DKIM signatures for incoming messages, see **Verify message DKIM header** in the “Message Characteristics” section.

In order for this action to take effect, you must configure `/opt/pmx6/etc/dkim.conf`. This configuration file contains the required signing options, and the location of the private key that is used to create the DKIM signature. If you want multiple PureMessage servers in the same domain to share one signature for the delivery of mail, you must also add the location of the private key to the **DomainKeys-Identified-Mail** publication. For configuration instructions, see the `dkim.conf` man page, and “Managing Publications” in the Server Groups Tab section of the *Manager Reference*.

DKIM signatures depend on access to a private key. This must also be configured in order for the action to take effect. For general information, see www.dkim.org.

The action appends a signature to messages, and, therefore, should only be applied to rules in the “Mail from internal hosts” section of the PureMessage policy.

Stop processing

When you add this action to a rule, it prevents processing of subsequent rules if that rule is hit. If you create a new rule by clicking add rule or Add Alternative, the Stop processing action is added by default.

Tempfail the message

Signal the MTA to return SMTP error code 421 (service not available).

Write an entry to pmx_log

Log a message to the PureMessage log file (as specified by the 'log_to' parameter in the `pmx.conf` configuration file).

- **Priority String:** Enter the priority of the log entry. The following strings can be used: DEBUG, INFO, NOTICE, WARNING, ERR, CRIT, ALERT, EMERG.
- **Message String:** Enter the text of the message to be logged.

Write test data to message log

This action is used for policy tests. When tests are run, this action writes a marker to the message log, and uses that marker to track the test message's progress through the policy filter. This action must be defined in the policy in order to run policy tests.

Related concepts

[Message Characteristics](#) (page 81)

[Managing Publications](#) (page 184)

Related tasks

[Creating Lists or Maps](#) (page 114)

Related information

[pmx-policy](#)

[pmx-queue](#)

[dkim.conf](#)

www.dkim.org

Template Variables

Certain predefined variables are available for use in banners and headers. You can choose from the predefined variables listed below, or create variables using the **Set a template variable** action.

The following variables can be used at any time:

- **%%PMX_VERSION%%:** The version of PureMessage.
- **%%SUBJECT%%:** The subject of the message. If there are multiple Subject headers, only the last occurrence is used.

- `%%MESSAGE_SIZE%%`: The size of the message, in bytes.
- `%%HEADER_SIZE%%`: The size of the message header, in bytes.
- `%%BODY_SIZE%%`: The body size in bytes.
- `%%QUEUE_ID%%`: The mail transfer agent's queue ID.
- `%%SENDER_IP%%`: The sender's IP address.
- `%%DATETIME%%`: A string containing the local date and time (for example, Thu Apr 24 12:49:28 2003).
- `%%DATETIME_GMT%%`: A string containing the GMT date and time (for example, Thu Apr 24 12:49:28 2003).
- `%%ENVELOPE_TO%%`: A comma-separated list of the envelope recipients.
- `%%ENVELOPE_FROM%%`: The envelope sender.
- `%%HEADER_FROM%%`: The From field of the message header.
- `%%HEADER_TO%%`: The To field of the message header. All occurrences of the To field are returned in a comma-separated list.
- `%%HEADER_CC%%`: The Cc field of the message header. All occurrences of the Cc field are returned in a comma-separated list.
- `%%HEADER_DATE%%`: The Date field of the message header.

The following variables can only be used after a spam probability test has been performed. The GAUGE variables use non-numeric characters for spam probability percentages. This is because many client mail programs cannot perform numeric comparisons.

- `%%GAUGE%%`: Subject-style gauge. At least one '#' is always appended, indicating that the message contains 0-50% spam. An extra '#' is added for every 10% above the argument that is given to `pmx_spam_prob`.
- `%%XGAUGE%%`: Absolute gauge, one 'X' character for every 10% probability.
- `%%SGAUGE%%`: Absolute gauge, one '*' character for every 10% probability.
- `%%IGAUGE%%`: Absolute gauge, one 'I' character for every 1% after the XGAUGE level.
- `%%PROB%%`: Spam probability (NN%).
- `%%HITS%%`: A listing of all the rules that were found by the spam engine. Custom rules names are appended with a plus symbol; modified default rule names are appended with an exclamation mark. For example:

```
MY_CUSTOM_RULE+ 4, OVERRIDDEN_RULE! 2, DEFAULT_RULE 1.1
```

- `%%SPAM_REPORT%%`: A verbose listing of the rule names, rule descriptions and rule scores for each rule triggered by the message. Custom rules names are appended with a plus symbol; modified default rule names are appended with an exclamation mark. For example:

```
MY_CUSTOM_RULE+      4.000  Site rule
OVERRIDDEN_RULE!    2.000  Default rule description
DEFAULT_RULE        1.100  Default rule description
```

The following variables are available after either the **Message is from blocked IP** or **Message contains a virus** test has run.

- `%%BLOCKLIST_REASON%%`: The reason, specified in the Sophos Labs blocklist data, for blocking a message based on IP. It is the same as the "reply" string seen in the SMTP session. Available after the **Message is from blocked IP** test has been run on a message.
- `%%VIRUS_IDS%%`: IDs of viruses detected in the message (for example, 'W32/Klez.h@MM'). Available after the **Message contains a virus** test has been run on a message.

The following variables are available inside rules containing specific tests or actions:

- **%%ATTACHMENT_NAME%%**: The name of the attachment. Available inside rules using the **Rename attachment** and **Replace body** actions.
- **%%ATTACHMENT_NAMES%%**: A list of all attachments. Available inside rules using the **Attachment name**, **Attachment size** or **Attachment type** tests or the **Rename attachment** action.
- **%%ATTACHMENT_SIZE%%**: The size of the attachment. Expanded by the **Replace message part** action.
- **%%ATTACHMENT_TYPE%%**: The content-type of the attachment (as given by the Content-Type header). Expanded by the **Replace message part** action.

Adding a New Rule

1. On the default page of the **Policy** tab, click **add main rule** at the bottom of the page.
This creates a new rule at the same hierarchical level as "Mail from internal hosts" and "Mail from external hosts" in the default PureMessage policy script.
2. In the **(New Rule)** text box, type a descriptive title.
3. In the **<select the test>** drop-down list, click the test that you want to use.

Depending on the test that you choose, additional text boxes may be displayed, which must be selected or filled, as appropriate. See "About Tests" for more information on these choices.

Note

Optionally, you may add multiple tests by clicking **add test**, or remove unwanted tests by clicking the delete icon [x] immediately below the row for the test that you want to delete.

4. In the **actions** drop-down list, select the action that you want to use.
Depending on the action that you choose, additional text boxes may be displayed, which must be selected or filled, as appropriate. See "About Actions" for more information on these choices.

Note

Optionally, you can add rules as part of the action taken, add multiple actions by clicking **add test**, or remove unwanted rules or tests by clicking the delete icon [x] immediately below the row for the rule or test that you want to delete. You can also change the order of multiple tests by clicking the up or down arrow icons immediately below the appropriate row.

5. **Save**, **Cancel**, **Copy**, **Cut**, or **Delete** the new rule or **Add Alternative** rule as appropriate.

While the policy is being edited, the edits are stored in a temporary file (`policy.siv.edit`). The changes are not made active until you **Commit** them. The following buttons are available when a rule is selected for editing:

- **Save**: Writes the changes to the temporary policy file, `policy.siv.edit`.
- **Cancel**: Removes the changes without writing to the temporary policy file.
- **Copy**: Copies the current rule (along with its associated tests and actions), so that it may be pasted elsewhere in the policy script.
- **Cut**: Removes a rule (along with its associated tests and actions) from its current position, so that it may be pasted elsewhere in the policy script.
- **Delete**: Deletes the entire rule, including the associated tests and actions.

- **Paste:** Inserts the most recently cut or copied rule immediately after the currently selected rule. The new rule is displayed at the same level of the hierarchy as the preceding rule. The rule is added as an "IF" rule or an "ELSEIF" rule, depending on the context. If you click the **Paste** button in a rule that is the last in a series of nested rules (an "ELSE" rule), or if it is a stand-alone "IF" rule, then a new "IF" rule is added. However, if you click the **Paste** button in a rule that is an "ELSEIF" or "IF" rule, followed by either an "ELSEIF" or "ELSE" rule, then a new "ELSEIF" rule is added.
- **Add Alternative:** Multiple rules can be specified at the same hierarchical level of the policy. When a second rule is specified using the **Add Alternative** button, it creates an "or" condition between the new rule and the rule above. Therefore, the second rule is only executed if the test in the first rule is not true. When rules are added using the **add rule** button, it creates an "and" condition between the new rule and the rule above. Regardless of whether the test in the first rule is true, subsequent rules will be processed.

Note

The **Copy**, **Cut** and **Paste** options are significant because of the importance of rule sequence. See "Policy Rules: Order of Processing" for more information.

6. **Commit, Revert, or generate a Source Diff.**

While the policy is being edited, the edits are stored in the temporary `policy.siv.edit` file. When you think that you have the policy changes made to your satisfaction, you can perform any of these actions:

- **Commit:** Writes the temporary policy file (`policy.siv.edit`) to the "live" policy file (`policy.siv`).

Note

It is strongly advised that you test the modified policy before you commit it. See "Testing the Current Policy" for information on this procedure.

- **Revert:** Deletes the temporary policy file (`policy.siv.edit`) and returns to the display of the unaltered policy file (`policy.siv`).
- **Source Diff:** Displays a comparison of the source code of the temporary policy file that contains the changes with the current "live" policy file.
- **see the source:** Displays the source code of the temporary policy file that contains the changes. To edit the source code, click the filename. To return to the graphical policy editor, click **go to constructor mode**.

Related concepts

[About Tests](#) (page 81)

[About Actions](#) (page 93)

Related tasks

[Policy Rules: Order of Processing](#) (page 80)

[Testing the Current Policy](#) (page 112)

Editing an Existing Rule

1. On the default page of the **Policy** tab, click the name of the rule that you want to edit.

The **Rules** page is re-displayed with blank editable text boxes.

2. In the **<rule name>** text box, change the descriptive title as required. This is title displayed in the rule tree on the default page of the **Policy** tab.
3. In the **<test>** drop-down list, select a different test if a change is wanted.

Depending on the test that you choose, additional text boxes may be displayed, which must be selected or filled, as appropriate. See "About Tests" for more information on these choices.

Note

Optionally, you can add multiple tests by clicking **add test**, or remove unwanted tests by clicking the delete icon [x] immediately below the appropriate row. You can also change the order of multiple tests by clicking the up or down arrow icons immediately beneath the appropriate row.

4. In the **actions** drop-down list, select the action that you want to use.

Depending on the action that you choose, additional text boxes may be displayed, which must be selected or filled, as appropriate. See "About Actions" for more information on these choices.

Note

Optionally, you may add rules as part of the action taken, or add multiple actions by clicking **add test** or remove unwanted rules or tests by clicking the delete icon [x] immediately below the row for the rule or test that you want to delete.

5. **Save**, **Cancel**, **Copy**, **Cut**, or **Delete** the modified rule or **Add Alternative** rule as appropriate.

While the policy is being edited, the edits are stored in a temporary file (`policy.siv.edit`). The changes are not made active until you **Commit** them. The following buttons are available when a rule is selected for editing:

- **Save**: Writes the changes to the temporary policy file, `policy.siv.edit`.
- **Cancel**: Removes the changes without writing to the temporary policy file.
- **Copy**: Copies the current rule (along with its associated tests and actions), so that it may be pasted elsewhere in the policy script.
- **Cut**: Removes a rule (along with its associated tests and actions) from its current position, so that it may be pasted elsewhere in the policy script.
- **Delete**: Deletes the entire rule, including the associated tests and actions.
- **Paste**: Inserts the most recently cut or copied rule immediately after the currently selected rule. The new rule is displayed at the same level of the hierarchy as the preceding rule. The rule is added as an "IF" rule or an "ELSEIF" rule, depending on the context. If you click the **Paste** button in a rule that is the last in a series of nested rules (an "ELSE" rule), or if it is a stand-alone "IF" rule, then a new "IF" rule is added. However, if you click the **Paste** button in a rule that is an "ELSEIF" or "IF" rule, followed by either an "ELSEIF" or "ELSE" rule, then a new "ELSEIF" rule is added.
- **Add Alternative**: Multiple rules can be specified at the same hierarchical level of the policy. When a second rule is specified using the **Add Alternative** button, it creates an "or" condition between the new rule and the rule above. Therefore, the second rule is only executed if the test in the first rule is not true. When rules are added using the **add rule** button, it creates an "and" condition between the new rule and the rule above. Regardless of whether the test in the first rule is true, subsequent rules will be processed.

Note

The **Copy**, **Cut** and **Paste** options are significant because of the importance of rule sequence. See “Policy Rules: Order of Processing” for more information.

6. **Commit, Revert**, or generate a **Source Diff**.

While the policy is being edited, the edits are stored in the temporary `policy.siv.edit` file. When you have finished making changes to the policy, you can perform any of these actions:

- **Commit:** Writes the temporary policy file (`policy.siv.edit`) to the "live" policy file (`policy.siv`).

Note

It is strongly advised that you test the modified policy before you commit it. See “Testing the Current Policy” for information on this procedure.

- **Revert:** Deletes the temporary policy file (`policy.siv.edit`) and returns to the display of the unaltered policy file (`policy.siv`).
- **Source Diff:** Displays a comparison of the source code of the temporary policy file that contains the changes with the current "live" policy file.
- **see the source:** Displays the source code of the temporary policy file that contains the changes. To edit the source code, click the filename. To return to the graphical policy editor, click **go to constructor mode**.

Related concepts

[About Tests](#) (page 81)

[About Actions](#) (page 93)

Related tasks

[Policy Rules: Order of Processing](#) (page 80)

[Testing the Current Policy](#) (page 112)

2.2.4 Managing the Repository

By default, PureMessage ships with a repository of policy code snippets, which is accessed by clicking **Policy Repository** on the sidebar of the **Policy** tab.

You can add these snippets to the PureMessage policy. You can also add new snippets to the repository, which can then be renamed, edited, and deleted as necessary.

About Default Policy Snippets

The **Policy Repository** contains nine snippets, which are described in the sections that follow. Policy snippets provide a convenient way to add new logic to the PureMessage policy script. The **Policy Repository** features of the PureMessage Manager make it easy to add these blocks of code to the policy script.

Adding a Disclaimer

Many sites require all outgoing email to have a legal disclaimer attached. PureMessage can centralize this operation at the mail gateway. This snippet contains the following code:

```
/* Detect outgoing mail */
if pmx_relay :memberof "internal-hosts" {
  pmx_add_banner :body :use_html_pre :file "banner.txt";
}
```

The `pmx_relay` test returns true if the email originates from an internal host. PureMessage typically treats mail from internal hosts as "outbound" mail. The `pmx_add_banner` action adds the contents of the file "banner.txt" to the body of the message. For HTML messages, it wraps the file in `<PRE>` tags, which makes the banner look more like plain text. If the message does not contain any text, the banner is added as an attachment at the end of the message.

Adding a Header

It is sometimes useful to add a header to all messages. For example, you may want to add a header in order to track messages. This snippet adds an X-Seen-By header to all messages. It contains the following code:

```
pmx_add_header "X-Seen-By" "%%HOSTNAME%%";
```

The `pmx_add_header` action adds an X-Seen-By header. The value of the header is the hostname of the PureMessage machine (for example, mail.example.com).

The `%%HOSTNAME%%` token is called a "template variable". For a list of supported template variables, see the `pmx-policy` man page.

Unlike the other default snippets in the Policy Repository, this snippet must be modified to suit its specific purpose. Occasionally, a site may require that certain messages be automatically sent to a specified address. For example, all messages containing the keyword "bug" might be sent to a bug-tracking system.

This example snippet adds a new recipient (bugs@example.com) if (a) the Subject has "bug" in it, (b) a recipient is "old-bugs@example.com", or (c) the header X-Bug-Id exists.

```
if anyof (header :matches "Subject" "*bug*", envelope "to"
          "old-bugs@example.com", header :matches "X-Bug-Id" "")
{ pmx_add_recipient
  "bugs@example.com"; }
```

The `anyof` test is a "meta-test"; it returns true if any of the tests return true.

The header test returns true if the Subject matches `*bug*`. The envelope test returns true if any of the message's "to" recipients match "old-bugs@example.com". Neither of these tests is case sensitive.

The second header test returns true if the X-Bug-Id header matches anything at all. If the X-Bug-Id header exists, this will return true. If any of the tests return true, then `pmx_add_recipient` causes the message to be delivered to "bugs@example.com", in addition to the original recipients.

Related information

[pmx-policy](#)

Catching Viruses

PureMessage is commonly used to protect an enterprise from viruses. Use the following code to create a rule that quarantines messages containing viruses and informs the recipients:

```
if pmx_virus {
    pmx_file "Virus";
    pmx_virus_clean "cantclean.tmpl";
    pmx_replace_header "Subject" "[PMX:VIRUS] %%SUBJECT%%";
}
```

Although this rule copies the message to PureMessage's quarantine, it does not block delivery of the message. Instead, it attempts to clean the virus-laden message. If the message was successfully cleaned, the message continues on to its original recipients. If the virus cannot be cleaned, the infected part is replaced with the error template `cantclean.tmpl`. The Subject is marked so recipients can plainly see that PureMessage found a virus. The `cantclean.tmpl` is the default template. You can create your own template based on it.

Default Internal Hosts Treatment

Although mail from external hosts is the primary concern for most sites, it is also common to establish a policy for internal hosts. For example, if you want to create a rule for internal hosts that rejects all messages containing a virus, use the following code:

```
if pmx_relay :memberof "internal-hosts" {
    pmx_mark1 "i";

    if pmx_virus {
        reject "One or more viruses were detected in the message.";
        stop;
    }
}
```

If the `pmx_relay` test returns true (that is, if the hostname or IP address of the server matches an entry in the "Internal hosts" list), the message is marked with an "i", which is added to the message log to indicate that this policy rule was hit.

If the `pmx_virus` test returns true, the message is rejected for the reason that it contains one or more viruses.

Some sites choose to implement a rule that accepts mail from hosts and senders included in the Whitelisted hosts or Whitelisted senders lists. This snippet contains the following code:

```
if anyof(pmx_relay :memberof "whitelisted-hosts",
        envelope :memberof "From" "whitelisted-senders",
        envelope :memberof "From" "whitelisted-senders-per-user")
{
    keep;
    stop;
}
```

This rule accepts a message and stops processing if (a) the hostname or IP address matches an entry in the list of Whitelisted hosts, (b) the contents of "Envelope from" are included in the **Whitelisted senders** list, or (c) the contents of "Envelope from" are included in the **Whitelisted senders (per-user)** list.

Detecting Spam

The most common use for PureMessage is detecting spam. This snippet contains the following code:

```
if pmx_spam_prob :over 50 {
    pmx_replace_header "Subject" "[SPAM:%%GAUGE%%] %%SUBJECT%%";
    pmx_add_header "X-PMX-Spam" "Probability=%%PROB%%";
}
```

If a message's spam probability is greater than 50%, PureMessage will prefix the Subject header with a string like [SPAM:###], where each additional # character denotes 10% above the argument to pmx_spam_prob. Subject-style gauge. At least one '#' is always appended, indicating that the message contains 0-50% spam. In this example, a message with a spam probability of 60% would have its subject prefixed with [SPAM:##]. A message with only 50% probability would have its subject prefixed with [SPAM:#].

Also, if PureMessage deems the message to be spam, it adds an X-PMX-Spam header, which spells out the probability in full. The header might look like this:

```
X-PMX-Spam: Probability=63%
```

Quarantining Spam Messages

Many sites prefer to quarantine messages that are more "spammy" than a given threshold to reduce the volume of messages in their users' mailboxes. This snippet contains the following code.

```
if pmx_spam_prob :over 80 {
    pmx_quarantine "Spam";
    stop;
}
elseif pmx_spam_prob :over 50 {
    pmx_replace_header "Subject" "[SPAM:%%GAUGE%%] %%SUBJECT%%";
    pmx_add_header "X-PMX-Spam" "Probability=%%PROB%%";
}
```

Messages with a spam probability greater than 80% will be quarantined immediately. Messages with a spam probability between 50% and 80% will be delivered, but with their subjects marked and an X-PMX-Spam header added.

The pmx_quarantine action copies the message to the PureMessage quarantine and prevents the original message from being delivered. The "Spam" argument is the reason for quarantining the message. This string should be a single word (if you use any spaces, they are silently changed to underscores).

Messages that have been quarantined can be viewed, released and added to a digest using features on the **Quarantine** tab in the PureMessage Manager.

Reject Messages Above 100K in Size

Many sites set a maximum size for email messages. Enforce this policy in the PureMessage policy with the following code:

```
if size :over 100K {
    reject "Message is too large: maximum 100K";
    stop;
}
```

The size test returns true if the message exceeds 100K. To specify a size that is below a specified threshold, use :under instead of :over.

Adding a Snippet to the Policy

1. Click the name of the snippet (for example, **Adding A Disclaimer**).
The Repository File page for that snippet is displayed.
2. Click **Copy To Clipboard**.
The **Policy Repository** page is displayed.
3. On the **Policy** tab sidebar, click **Policy Rules**.
The **Rules** page is displayed.
4. Click the rule that comes immediately before the desired location for the new rule.
For example, Click **Clean mail containing viruses** if you want the new rule to be inserted after that rule.
5. Click **Paste**.
The pasted snippet will be displayed as a rule in the PureMessage policy.

Related tasks

[Adding a Snippet to the Repository](#) (page 110)

Adding a Snippet to the Repository

Rules that are part of the PureMessage policy can be copied and saved to the Policy Repository as a snippet.

To add a snippet to the Policy Repository:

1. On the **Policy** tab sidebar, click **Policy Rules**.
2. Click the rule that you want to save.
3. Click **Copy**.
4. On the **Policy** tab sidebar, click **Policy Repository**.
5. Click **Clipboard**.
The copied rule is displayed on the Clipboard.
6. Click **Save to Repository**.
A **Repository File** page for the new snippet is displayed with a PureMessage-generated name (for example, "clip000").
7. Rename the snippet, and click **Update**.
It is displayed in the list of snippets in the Policy Repository.

Related tasks

[Adding a Snippet to the Policy](#) (page 110)

Changing the Name/Description of a Snippet

1. Click the name of the snippet (for example, **Adding A Disclaimer**).
The **Repository File** page for that snippet will be displayed.
2. In the **Name** and/or **Description** fields, enter the new text.
3. Click **Update**.

Editing a Snippet with the Manager

Snippets cannot be edited through the **Policy Repository** interface. They must first be copied to the PureMessage policy. Edit snippets via the fields on the **Rules** page of the **Policy** tab, or by changing the source code for the policy script. After a rule has been edited, you have the option of adding the snippet to the Repository.

To edit a snippet using the Manager interface:

1. Click the name of the snippet that you want to edit (for example, **Adding A Disclaimer**).
The **Repository File** page for that snippet is displayed.
2. Click **Copy to Clipboard**.
The file will be copied to the Clipboard, which is displayed as an item at the top of the list of snippets in the Policy Repository.
3. On the **Policy** tab sidebar, click **Policy Rules**.
4. Click the rule that comes immediately before the desired location for the new rule.
For example, click **Clean mail containing viruses** if you want the new rule to be inserted after that rule.
5. Click **Paste**.
The pasted snippet will be displayed as a rule in the PureMessage policy.
6. Edit the "tests" and "actions" fields for a given rule to modify the snippet, and click **Save**.

Note

Complete the next two steps only if you want to add the rule to the policy script. Otherwise, follow the instructions for "Adding a Snippet to the Repository", and then revert changes to the policy script.

7. Click **Commit**.
8. Click **Restart now** to activate the changes.

Related tasks

[Editing a Snippet in the Source File](#) (page 111)

[Adding a Snippet to the Policy](#) (page 110)

Editing a Snippet in the Source File

Snippets cannot be edited through the **Policy Repository** interface. They must first be copied to the PureMessage policy. Edit snippets via the fields on the **Rules** page of the **Policy** tab, or by changing the source code for the policy script. After a rule has been edited, you have the option of adding the snippet to the Repository.

To edit a snippet in the source file:

1. Click the name of the snippet that you want to edit (for example, **Adding A Disclaimer**).
The **Repository File** page for that snippet is displayed.
2. Click **Copy to Clipboard**.
The file will be copied to the Clipboard, which is displayed as an item at the top of the list of snippets in the Policy Repository.
3. On the **Policy** tab sidebar, click **Policy Rules**.

4. Click **see the source** in the top right corner of the **Policy Rules** page.

A read-only version of the policy script's source code is displayed.

5. Click the file path link at the top of the policy script.

An editable version of the policy script is displayed.

6. Make the desired changes to the Sieve script, and click **Save**.

The read-only version of the script will be displayed again, along with messages advising that your changes have been saved but that they have not been committed.

Note

Complete the next two steps only if you want to add the rule to the policy script. Otherwise, follow the instructions for "Adding a Snippet to the Repository", and then revert changes to the policy script.

7. Click **Commit**.
8. Click **Restart now** to activate the changes.

Related tasks

[Editing a Snippet with the Manager](#) (page 111)

[Adding a Snippet to the Policy](#) (page 110)

Deleting a Snippet from the Policy Repository

1. Select the check box next to the name of the snippet that you want to delete.
2. Click **Delete**.
PureMessage displays a message confirming the delete.
3. Click **Yes Delete**.

2.2.5 Testing the Current Policy

Use the **Policy > Test Current Policy** page to verify that configured rules are operating as expected. For example, you may have added a sender to the **Blacklisted Senders** list and want to verify that PureMessage no longer accepts mail from this sender.

A policy must be committed before it can be tested. However, until the Milter service is restarted, the new policy will not be active. Therefore, a policy can be tested without making it the active policy. This allows you to restore a backup policy if you are not satisfied with the performance of the new policy.

To test the current policy:

1. On the **Policy** tab sidebar, click **Test Current Policy**.
The **Test Current Policy** page is displayed.
2. Fill in or select the settings for the text boxes in the **Specify test options and message sources** table.

The test options are:

- **Select Relay Type:** To test the configured policy, you must specify either the relay type or an IP address. To identify the message source based on the relay, select **Internal (localhost)** or **External** from the drop-down list.

- **... or IP address:** To identify the message source by the IP address instead of the relay, enter an IP address in this text box. This text box can also be used to emulate a host defined as a trusted relay, thus bypassing DNS tests. To bypass DNS tests, enter an IP address that is defined in the **Trusted Relay IPs** list.
- **Envelope From:** Optionally, specify that PureMessage tests the validity of the message source based on the address specified by the message's envelope from information instead of the message's from address.
- **Envelope To:** Optionally, specify that PureMessage tests the validity of the message source based on the address specified by the message's envelope to information instead of the message's to address.
- **Select sample messages:** PureMessage includes a variety of sample messages for testing rules specified in the policy. To test using sample spam messages, select **Infected (messages contain viruses)**, **Spam (messages are typical examples of spam)** or **Normal (messages are legitimate)**. Additional messages can be added to the test message set. Messages must be in "mbox" format, and must be stored in the subdirectories beneath `etc/data/samples/` (located beneath the PureMessage installation directory).
- **... or paste message source here:** Instead of testing with the sample messages or with a specific messagepmx file, type or paste the message source information into this text box. The format must adhere to "mbox" format. For example, specify a sender's address by typing `From:`, followed by the email address. When entering message source information, be sure to position the cursor at the beginning of the text box. Also be sure there are no blank lines or spaces before or after message source text. A blank line or a space indicates the end of the message header. Any text entered afterwards is read as if it were part of the body of the message. The following text appears in the message source text box by default:

```
To: PureMessage Test User <puremessage-test@ServerName.example.com>
From: PureMessage Admin <postmaster@ServerName.example.com>
Subject: PureMessage Test
Date: Wed, 31 Mar 2004 00:40:06 GMT

This message was generated to test PureMessage.
The current local time is: Tue Mar 30 16:40:06 2004

Have a nice day,
PureMessage Admin
```

3. When the desired test criteria is entered, click **Test**.

Test results are displayed as follows:

- **Original Message File Name:** If using a local message file or sample messages, the file name(s) are displayed in this column.
- **Resulting Message:** Every test message is written to the quarantine regardless of the resulting action. This column lists the quarantine ID number for the message. Click the number to view the message details.
- **Delivery Action:** This column displays the action that would have been taken if the message had been "live".
- **Details:** Displays debug messages for the test process.

2.2.6 Using Lists and Maps

In the **Lists** and **Address Maps** sections of the **Policy** tab are links that provide the ability to create, edit and manage lists and maps.

Creating Lists or Maps

The procedure for creating either a new list or a new map is mostly the same, and both procedures use the same page and form in the PureMessage Manager.

Lists are used in the policy to exclude or include the listed groups or addresses from tests or actions. For example, in the default policy, messages originating from addresses contained in the **Whitelisted hosts** list are not scanned for spam.

Maps are used to associate one email address with another for the purpose of redirection (as with the **Notifications Address Map**). For example, the policy could be configured to redirect messages from an alias email address to a personal email address via the use of an address map. Maps can also be used to apply user preferences (as with the **Recipient Aliases Map**), or, in the case of custom maps, for a user-defined purpose.

Note

Although PureMessage supports the creation of LDAP-based lists and maps, these lists and maps are read-only; you can only edit them using LDAP tools and not with PureMessage.

To create a new list or address map:

1. On the sidebar of the **Policy** tab, click **New** beside **Lists** to create a new list, or click **New** beside **Address Maps** to create a new address map.

The **Add List/Map** page is displayed, with either **List** or **Map** displayed in the **Type** drop-down list, depending on which **New** button you clicked.

2. In the **Create New List/Map** table, specify the following:

- a) **Type:** Ensure that the correct type is selected, either **List** or **Map**.

- b) **ID:** Enter the identity that will be used for the list's or map's filename.

Lists and maps are stored in individual configuration files. The name that you enter in this text box becomes the filename. The name that you enter should therefore describe the purpose of the list or map, and it should be in a format that is usable as a filename. For example, the **Anti-spam opt-outs** list has an identity of **anti-spam-optouts**.

- c) **Name:** Enter the name of the list or map.

This is the name that will be displayed on the **Policy** tab sidebar.

- d) **Description:** Enter a meaningful description for the list or address map.

This information will be displayed in the **Description** column of the **Configured Lists** or **Configured Maps** table that is displayed when you click **manage** at the bottom of either the **Lists** or **Address Maps** sections of the **Policy** tab sidebar.

- e) **Match Type:** Enter the matching method that you want the message filtering to use when the policy service compares a message to the contents of the list.

See "Match Types" for detailed information on the significance of each of these choices.

3. In the **Source** section, select either **Flat file** or **LDAP**. If you select **LDAP**, you must also enter the following information:

- a) **LDAP Server:** Specify the 'host:port' of the server(s) to connect to when authenticating users via LDAP.

To specify more than one LDAP host for failover, enter a list of hosts separated by semicolons. If no ':port' is specified, port 389 is used by default. To use an encrypted LDAPS connection,

simply prefix the host:port with 'ldaps://'. For LDAPS connections, port 636 is used if no port is specified. For example:

```
localhost:636 ldaps://ldap.mycompany.com
```

As a failover, it is strongly advised that you specify two or more LDAP servers in the LDAP Server text box. This is done by separating the URLs with semicolons. For example:

```
ldap://myhost1:389;ldap://myhost2:389;ldap://myhost3:389
```

- b) **DN for binding to LDAP server:** Specify the Distinguished Name(DN) used to connect to the LDAP server in order to query the Distinguished Name of the user the system is attempting to authenticate. This text box supports variable substitution.
- c) **Password for binding to LDAP server:** Specify the password used to connect to the LDAP server in order to query the Distinguished Name (DN) of the user that the system is attempting to authenticate. This DN and password should be granted minimal rights but must be able to perform a query to retrieve the DN for a user based on their provided username/id.

Note

Any password entered when adding an LDAP-based list or map appears in a plain text file, either `etc/maps.conf` or `etc/lists.conf`. It is therefore suggested that you use the password for an LDAP account with privileges limited to user and password authentication.

- d) **Base DN for matching:** Specify the top LDAP directory node underneath which the search is performed to retrieve the DN of the user. This text box supports variable substitution.
 - e) **Filter to match list/map:** Specify the LDAP or Active Directory search string. This text box supports variable substitution. To match on an email address, use the `%s` variable. You can also use `%u` to match only on the portion of the address to the left of the `@` symbol, or `%d` to match everything to the right of the `@` symbol.
 - f) **Map attribute:** When creating a new map, enter the 'map to' value here. For example, if converting user IDs to email addresses, the map attribute could be 'mail'.
4. Once all the required information is set, click **Save** to add the list or map.

Related concepts

[Match Types](#) (page 115)

Related tasks

[Editing Lists](#) (page 120)

[Editing Maps](#) (page 121)

[Managing Lists and Maps](#) (page 122)

Related information

[Variable substitution](#)

Match Types

The **Match Type** is the matching method that is used by the policy engine when it compares a message to the contents of the list. The **Match Type** is set when creating lists or maps, and it can be changed when managing lists or maps.

The available match types are:

- **Exact:** The item in the list or map must exactly match the item to which it is compared. When creating an LDAP-based list or map, only this option is available.
- **Email Globbs:** Glob style email address matching. Allowable wildcards are:

- ? matches a single character, but not "."
- * matches a sequence of characters, but not "."
- ** matches a sequence of characters, including "."

"@" is prevented from matching on "*" and "?". If the list entry ends with "@" then the match is not anchored at the end. For more information, see "Matching Email Addresses" in the **Wildcard Usage** section.

Note

When creating an LDAP-based list, only the **Exact** option is available.

- **Email Segments (lists only):** This match type is more efficient than the "Email Globs" match type when performing the most common email address lookups. For example:

```
someuser@sophos.com
```

```
someuser@
```

```
@sophos.com
```

- **Substring (lists only):** Allows for a partial match between the list item and the item to which it is compared.
- **Glob (lists only):** Glob style matching. The wildcards are "*" (which matches any sequence of characters) and "?" (which matches a single character). A literal "*", "?" or "\" can be matched by escaping the character with a "\" (e.g "\" matches a literal "?").
- **Hostname and IP Masks (lists only):** Glob style domain and IP4 address matching. Matching is always case insensitive. If an IP address is followed by a "/" and a numeric value, then it is read as an IP4 address with a mask. In the examples that follow, <digits> represents an integer in the 0-255 range, <bits> represents an integer in the range of 0-32, and <mask> may consist of either an <IP> or <bits>. The entry format is as follows:

```
<ip4address> = <ip> [ "/"<mask>]
```

```
For example: 192.0.2.0/24
```

```
<ip> = <digits> "."<digits> "."<digits> "." <digits>
```

```
For example: 192.0.2.0
```

If the list entry does not start with a number, it is taken as a glob style string that will be matched against a domain name. Allowable wildcards are:

- ? matches a single character, but not "."
- * matches a sequence of characters, but not "."
- ** matches a sequence of characters, including "."

Matches are automatically anchored to the end of the string. A leading "@" can be used to force anchor to the beginning as well. If the list item starts with '!' then it is negated. For more information, see "Matching Hostnames and IP Addresses" in the **Wildcard Usage** section.

- **Regular Expression (lists only):** Lists can be configured to contain regular expressions. When creating a new list, specify the "Regular Expression" match type. Individual entries in the list are then entered as regular expressions. Regular expressions used within lists are not prefixed with

slashes or braces. Also, it is not necessary to escape special characters in regular expressions contained in lists, for example:

```
.*\.pif
.*\.doc\.exe
.*\.pif
your_details\.zi?
message\.zip
message\.zi
wicked_scr\.scr
wicked\.scr
.*\.scr
patch\.exe
sobig\.f\.txt
```

For more information about using regular expressions in lists and in the PureMessage policy, see the "Regular Expressions Primer".

- **Case Insensitive:** Select this check box to ignore the letter case used in the messages.

Related concepts

[Wildcard Usage](#) (page 117)

Related tasks

[Managing Lists and Maps](#) (page 122)

Wildcard Usage

The following sections describe using wildcards to match email addresses or hostnames in lists and maps.

Matching Email Addresses

- example.com matches all addresses that end with a domain name of "example.com", such as "foo@example.com", "foo@foo.example.com".
- @example.com matches all addresses with "example.com" as the exact domain part.
- dev-**@ matches all addresses that start with the string "dev-".
- foo@perl.* matches all addresses with "foo" as the local part and a two-level domain name with "perl" as the first level, like "foo@perl.com" and "foo@perl.org".
- foo@perl.** matches all addresses with "foo" as the local part and at least a two-level domain name with "perl" as the first level, like "foo@perl.com" and "foo@perl.domain.com".

Matching Hostnames and IP Addresses

Note

Some lists, such as IP Blocking Exceptions and Trusted Relays, only accept IP addresses, not domain names.

- 127.0.0.1 matches the exact IP address.
- 10.10.10.0/24 matches any address in the 10.10.10.0 network.
- example.com matches any hostname in the example.com domain as well as "example.com" itself. Does not match "notexample.com" or "example.com.org".
- **.example.com matches any hostname in the example.com domain but not "example.com" itself.
- .example.com matches any hostname in the example.com domain but not "example.com" itself.

- *.example.com matches hosts like "foo.example.com", but not multilevel names like "foo.bar.example.com".
- @example.com only matches the host "example.com" and not any subdomains.

To add items to lists, see “Editing Lists”. To add items to address maps, see “Editing Maps”.

Related tasks

[Editing Lists](#) (page 120)

[Editing Maps](#) (page 121)

About PureMessage Default Lists

This section provides descriptions of the lists that ship with PureMessage. It describes their use in the PureMessage policy, and it describes the **Match type** that is set for the list, which determines the form in which you can set non-specific entries. For more information, see “Match Types”.

PureMessage ships with the following default lists:

- **Anti-Spam Opt-Outs:** This list is used to exempt mail addressed to specific users from spam checks. Mail addressed to members of this list is immediately delivered to the recipients and does not undergo a spam check, or any tests and actions included in the Policy Rules. Spam-checking rules can also be configured to ignore mail destined for users defined in this list. Match type: Email Globs.
- **Blacklisted Hosts:** Blacklisted Hosts are relays known to distribute spam or viruses. Policy tests and actions can be configured, for example, to reject or quarantine messages originating from relays in this list. By default, this list is shared to other hosts in multi-server deployment as part of the Policy publication. Match type: Hostnames and IP Masks.
- **Blacklisted Senders:** Blacklisted senders are addresses known to distribute spam or viruses. Policy tests and actions can be configured, for example, to reject or quarantine messages originating from addresses in this list. By default, this list is shared as part of the Policy publication. Match type: Email Globs.
- **End Users:** Lists the addresses of the users who can access the End User Web Interface (EUWI). The default value grants all PureMessage end users permission to use the EUWI. Match type: Email Globs.
- **IP Blocking Exceptions List:** This list is used to define IP addresses and fully qualified hostnames that should be explicitly allowed by the IP Blocker Service (see “Enabling or Disabling MTA IP Blocking”) and the PureMessage policy. Entries in this list override blacklisted IP addresses in the data package from Sophos Labs. Match type: Hostnames and IP Masks.
- **IP Blocking Inclusions List:** This list contains the IP addresses and fully qualified hostnames that should be blocked by the IP Blocker Service (see “Enabling or Disabling MTA IP Blocking”) and the PureMessage policy. Entries in this list override whitelisted IP addresses and hostnames in the Sophos Labs data. This list must be added to a publication before it can be shared with other hosts in a multi-server deployment. Match type: Hostnames and IP Masks.
- **Internal Hosts:** Domain names or IP addresses configured in this list are assumed to be internal. This list can be used to exempt specific hosts from policy rules. By default, "127.0.0.1" (localhost) is added to this list. By default, this list is shared as part of the Policy publication. Match type: Hostnames and IP Masks.
- **Offensive Words:** A list of "restricted" words. The Offensive Words List can be used in a policy rule that quarantines messages if one of the words is found. This list must be added to a publication before it can be shared with other hosts in a multi-server deployment. Match type: Regular Expressions.
- **Quarantine Digest Users:** This list is used to identify users who will receive Quarantine Digests. Only users with messages in the quarantine receive digests. Match type: Email Globs.

Note

You can also create multiple Quarantine Digest Users lists, for example, if you want to send out the Quarantine Digests at different times for recipients in different time zones or if you want to send out digests of different quarantine reasons for different groups of end users. For instructions on how to do this, see “Creating Customized Quarantine Digest Users Lists” in the Administrator’s Reference.

- **RPC Hosts:** Lists the IP addresses of other PureMessage servers. Scheduled jobs that are set up during installation allow the central PureMessage server to push content to the other PureMessage servers. Listing PureMessage IP addresses shares pre-configured lists (such as 'Blacklisted Senders' and 'End Users'). This list must be added to a publication before it can be shared with other hosts in a multi-server deployment. Match type: Hostnames and IP Masks.
- **Suspect Attachment Names:** The policy script can be configured to perform actions based on the attachment names specified in this list. If you include the Message contains suspicious attachments test in a policy script, PureMessage searches the filename for the attachments defined as suspicious. This list must be added to a publication before it can be shared with other hosts in a multi-server deployment. Match type: Globbs.
- **Suspect Attachment Type:** The Message contains suspicious attachments test also searches the Content-Type and Content-Disposition headers for media types specified in the Suspect Attachment Type list. This list must be added to a publication before it can be shared with other hosts in a multi-server deployment. Match type: Globbs.
- **Trusted Relay IPs:** Trusted relays are mail-filtering hosts that are known to be safe. PureMessage uses a Trusted Relay IPs list to differentiate between unknown relays and "internal" relays (or trusted external relays). Relays with IP addresses within the 127.*.*, 192.168.*.* and 10.*.* blocks are always treated as internal relays. By default, the IP address of the first "external" relay is tested against the RELAY_IN_* group of anti-spam rules. All other external relays are tested against the RCVD_IN_* group of anti-spam rules.

All IP addresses of relays that are known to be safe, but are not included in the IP address blocks described above, should be added to the Trusted Relay IPs list. For example, if an ISP provides message-relay services for your organization, the IP address of the ISP’s mail server should be included in the Trusted Relay IPs list.

Once the Trusted Relay IPs list is populated, configure the **Disable non-relay checks?** option on the **Policy: Anti-Spam Options** page.

Note

Match type: Hostnames and IP Masks, but only IP addresses (not domain names) can be entered in this list.

- **Whitelisted Hosts:** Whitelisted Hosts are relays that are known to be safe. Policy tests and actions can be configured, for example, to exempt messages originating from relays in this list from spam checking. By default, this list is shared to other hosts in multi-server deployment as part of the Policy publication. Match type: Hostnames and IP Masks.
- **Whitelisted Senders:** Whitelisted senders are addresses that are known to be safe. Policy tests and actions can be configured, for example, to exempt messages originating from addresses in this list from spam checking. By default, this list is shared to other hosts in multi-server deployment as part of the Policy publication. Match type: Email Globbs. IP addresses can also be entered if (and only if) they are obtained from a message’s "envelope-from" part.

- **Log Reasons:** This list allows you to control which reasons appear in the **Reason** drop-down list that is used for building log search queries in the Groups Web Interface. If you want to be able to search for a reason that is not included by default, you must add it to this list. See “Adding and Deleting Custom Log Search Reasons” in the Administrative Groups section of the *Administrator’s Reference* for more information. This list must be added to a publication before it can be shared with other hosts in a multi-server deployment.

Related concepts

[Match Types](#) (page 115)

[Managing Publications](#) (page 184)

Related tasks

[Enabling or Disabling MTA IP Blocking](#) (page 178)

[Creating Customized Quarantine Digest Users Lists](#) (page 145)

Editing Lists

Lists are used with the PureMessage policy to specify that email addresses, IP addresses, file extensions, and other matching strings are included in or excluded from policy tests and actions. Policy rules often use membership in a particular list as the condition for performing (or not performing) an action. For example, in the default PureMessage Policy, messages from email addresses contained in the **Whitelisted Hosts** list are not scanned for spam.

To edit a list:

1. On the sidebar of the **Policy** tab, click the name of the list that you want to edit.

The **Edit List** page is displayed.

2. Modify the selected list by performing any of the following procedures:

- To add an item to the list, enter one or more items, with only one item per line, in the **Add items** text box, and click **Add** at the bottom of the form.

To enter a "negative" item, precede the item with an exclamation mark. Lists that consist of email addresses usually support glob-style address matching and wildcards. Lists that contain IP addresses usually support IP masks.

When you click **Add items**, the item is listed in the **List items** pane.

- If your list contains a large number of items, you can filter which items are displayed by entering one or more characters in the **Filter** text box and clicking **Filter**.
- To edit an existing list item:

- a) Select the check box beside the item that you want to edit, and click **Edit**.

The selected item is redisplayed in an editable text box.

- b) Make the changes you want and click **Save**.

The changed item is saved and redisplayed in its changed form.

- To delete an item from the list, select the check box beside the item that you want to delete, and click **Delete** at the bottom of the form.

The item is removed.

Related tasks

[Creating Lists or Maps](#) (page 114)

[Editing Maps](#) (page 121)

[Testing Lists or Maps](#) (page 121)

[Managing Lists and Maps](#) (page 122)

Editing Maps

Maps are used to associate one email address with another for the sake of redirection (Notifications Address Map), for the sake of applying user preferences (Recipient Aliases Map), or, in the case of custom maps, for a user-defined purpose.

To map addresses:

1. On the sidebar of the **Policy** tab, click the name of the map that you want to edit.

The **Edit Map** page is displayed.

2. Modify the selected map by performing any of the following procedures:

- To add an address mapping, enter the address that you want mapped in the **Map From** text box, enter the address that you want it mapped to in the **Map To** field, and click **Add** at the bottom of the form.

Note

Addresses can be mapped to "empty" values. If you leave the **Map To** text box blank, this will have the result of deleting a recipient.

The mapping is listed below the edit text boxes, with the information you entered displayed.

- If your map contains a large number of entries, you can filter which entries are displayed by entering one or more characters in the **Filter** text box and clicking **Filter**.
- To edit an existing mapping:

- a) Select the check box beside the mapping that you want to edit, and click **Edit**.

The selected mapping is loaded into the **Map From** and **Map To** text boxes.

- b) Make the changes you want and click **Save**.

The mapping is listed below the edit text boxes, with the information you added displayed.

- To delete a mapping from the list, select the check box beside the mapping that you want to delete, and click **Delete** at the bottom of the form.

The mapping is removed.

Related concepts

[Address Maps](#) (page 253)

Related tasks

[Creating Lists or Maps](#) (page 114)

[Editing Lists](#) (page 120)

[Testing Lists or Maps](#) (page 121)

[Managing Lists and Maps](#) (page 122)

Testing Lists or Maps

The **Policy > Test Lists/Maps** page lets you test a policy list or map against a specific value.

Testing a List

To test a new or modified list:

1. On the sidebar of the **Policy** tab, click **Test List/Map**.

The **Test Lists/Maps** page is displayed.

2. In the **Test List/Map** table, select the list that you want to test from the **List Name** drop-down list.
3. In the **Value to test** text box, enter the value that you want to test.
4. Click **Test**.

The test will return either **Match** or **No Match**.

Testing a Map

To test a new or modified map:

1. On the sidebar of the **Policy** tab, click **Test List/Map**.

The **Test Lists/Maps** page is displayed.

2. In the **Test Map** table, select the map that you want to test from the **Map Name** drop-down list.
3. In the **Value to test** text box, enter the value that you want to test.
4. Click **Test**.

The test will return either **Match** or **No Match**.

Managing Lists and Maps

Click **Policy > Lists > manage** or **Policy > Maps > manage** to manage the existing lists and maps. These lists can be modified or deleted. You can:

- change the **Name** of the list or map
- change the **Description** of the list or map
- change the **Match Type** of the list or map (see “Match Types” for detailed information on the significance of each of these choices)
- change whether the list or map entries are **Case Insensitive**

Note

While the **Match Type** can be changed for new or default maps, or for new lists, you cannot change the **Match Type** of default lists.

To manage an existing list or map:

1. On the sidebar of the **Policy** tab, click **manage** at the bottom of the **Lists** or **Maps** sections.

The **Manage System Lists** page is displayed.

2. Perform one of the list or map management tasks:

- To modify a list or map:

- a) Click the name of the list or map.

The selected list or map information is displayed on either an **Edit List** or an **Edit Map** page.

- b) Make the changes that you want to the editable controls, and then click **Save**.

The list or map changes are saved and you are returned to the **Manage System Lists** page or the **Manage System Maps** page.

- To delete a list or map:

- a) Select the check box beside the name of the list or map that you want to delete, and then click **Delete**.

The selected list or map is removed from the **Manage System Lists** page or the **Manage System Maps** page.

Related concepts

[Match Types](#) (page 115)

2.2.7 Using the Anti-Virus and Anti-Spam Settings

The **Manage** section on the **Policy** tab sidebar includes links to three pages where you can set the behavior of Anti-Spam and Anti-Virus filtering:

- The **Anti-Virus Options** page allows you to set options such as whether pre-analysis of the message contents is done to save processing time, whether virus scanning is run as a system service, what the behavior should be when the anti-virus engine fails, and how the anti-virus engine should handle archive scanning.
- The **Anti-Spam Rules** page allows you to set which anti-spam rules are enabled or disabled and what the relative weight and probability adjustment percentage is for each rule. You can also create new anti-spam rules.

Note

Sophos does not recommend adjusting anti-spam rules because they are automatically updated by [SophosLabs](#) on a regular basis. It is also advised that you consult with Sophos support before attempting to create any new anti-spam rules. If you are receiving false positives or false negatives, Sophos asks that you forward these messages to SophosLabs for our analysts to investigate. See "PureMessage Feedback" in the Contacting Sophos section for more information.

- The **Anti-Spam Options** page allows you to set how much of a message should be scanned, whether non-relay checks should be enabled or disabled, whether network checks are enabled or disabled, and what special language character sets are enabled.

Related concepts

[Contacting Sophos](#) (page 74)

Setting Anti-Virus Options

1. On the sidebar of the **Policy** tab, click **Anti-Virus Options**.

The **Anti-Virus Options** page is displayed.

2. In the **Anti-Virus Options** table, change any of the following text boxes as required:
 - **Enabled:** Select this check box to enable virus-scanning using the Sophos engine.
 - **Quick Scan:** Quick scan analyzes the message for file types (and areas within those files) that can harbor viruses, thus reducing the amount of the message that needs to be scanned. Full scan performs the same tests as quick scan; in addition, it scans each byte of the message for patterns that match known viruses. Because email message sizes are generally quite small, full scan is recommended. The Sophos virus engine will stop scanning the message when a virus is found, regardless of this setting. This option is enabled by default.
 - **Run As Service:** Specify whether the anti-virus engine should be run as a service ("daemon") or should be loaded by the PureMessage engine. If run as a service, the Anti-Virus service can be started and stopped as a separate component on the **Local Services** tab of the PureMessage Manager. Configure the service's operating parameters on the **Anti-Virus Service Options** page.

- On Error: Specify the action that should be performed if the virus engine fails. Set this value based on whether it is more important to prevent viruses (Temp Fail) or deliver messages (Pass Through).
 - Scan Archives: If this option is enabled, the anti-virus scanner unpacks and scans all supported archive types. To enable or disable the scanning of specific archive types, edit the `sophos.conf` configuration file.
 - Scan Self-Extracting Archives: If this option is enabled, the anti-virus scanner searches inside PKLite, LZEXE and Diet archives.
3. Once you have made the required changes in the **Anti-Virus Options** table, click **Save**.
A message box appears, informing you that your changes have been saved.
 4. Click **OK**
You are returned to the **Rules** page.

Managing Anti-Spam Rules

PureMessage is distributed with a pre-configured set of anti-spam rules used to identify spam messages. Updates containing refined anti-spam *heuristics* are published frequently.

Rules are grouped together by function into feature groups. These groups are enabled or disabled from the **Anti-Spam Options** page. The individual rules are enabled or disabled on the **Anti-Spam Rules** page.

You can enable and disable existing rules, or create custom rules.

Note

Sophos does not recommend adjusting anti-spam rules because they are automatically updated by SophosLabs on a regular basis. It is also advised that you consult with Sophos support before attempting to create any new anti-spam rules. If you are receiving false positives or false negatives, Sophos asks that you forward these messages to SophosLabs for our analysts to investigate. See “PureMessage Feedback” in the Contacting Sophos section for more information.

Related tasks

[Configuring Anti-Spam Options](#) (page 127)

Viewing Anti-Spam Rules

1. On the sidebar of the **Policy** tab, click **Anti-Spam Rules**.
The **Anti-Spam Rules** page is displayed.
2. Optionally, filter the listed rules by doing one or more of the following:
 - Select the **Feature Group** that you want to view. The options are:
 - All Groups: Select this option to view all feature groups.
 - Spam Signatures Analysis: Rules designed to test specific parts of an email message (for example, the message body or attachments).
 - Known Spam Destinations: Rules that check URIs contained in a message against a list of known spam destinations.
 - Sender Reputation: Rules that call on network services (for example, DNS whitelists and blacklists) to check if the sender or relay is reputable.
 - Heuristic Analysis: Rules that use regular expressions to detect spam-like words, phrases, characters or patterns in messages.

- Site Features: Custom rules that test message content using regular expressions.
- Select the rule type that you want to view from the **Show** drop-down list. The options are:
 - All Rules: Select this option to view all rules.
 - Default Rules: Select this option to view only rules without a modified **Weight** or **Probability Adjust %**.
 - Overridden Rules: Select this option to view only rules with a modified **Weight** or **Probability Adjust %**.

Note

Site rules (default rules) are indicated by an "S" in the icon to the left of the rule; overridden rules are indicated by an "O".

- Select the rule state that you want to view from the **Rule State** drop-down list. The options are:
 - Any State: Select this option to view rules of any state.
 - Enabled: Select this option to view only enabled rules.
 - Disabled: Select this option to view only disabled rules.

Note

Disabled rules are indicated by a red dot in the icon to the left of the rule; enabled rules are indicated by a green dot.

3. Once all query parameters have been set, click **Filter** to the right of the filter text box to apply all filter criteria.

Related concepts

[Test Types](#) (page 273)

[Modifying Anti-Spam Rules](#)

Note

Only the **State**, **Weight**, and **Probability Adjust %** of default anti-spam rules can be modified. Sophos does not recommend adjusting anti-spam rules because they are automatically updated by SophosLabs on a regular basis.

To modify default anti-spam rule behavior:

1. On the sidebar of the **Policy** tab, click **Anti-Spam Rules**.

The **Anti-Spam Options** page is displayed.

2. Click the name of the rule that you want to modify.

The editable settings for the selected rule are displayed in drop-down lists or editable text boxes.

3. Make the desired modifications.

The editable settings and their available options are:

- Rule State: Select **Auto**, **Enabled**, or **Disabled**.

Note

By default, rules are set to **Auto**, which sets the state of the rule according to whether there is a value in the rule's **Weight** or **Probability Adjust %** text box. If both scores are zero, the rule has no effect.

- **Weight:** The value (or "weight") added to the message's total spam score when the message matches this rule. Values can be either positive or negative; prefix negative numbers with a minus symbol. For more information about how scores are calculated, see "Test Scores" in the Policy section of the *Administrator's Reference*.
 - **Probability Adjust %:** The absolute probability for the rule in the form of a percentage. When the total spam score is calculated for the message, rules with weights are first converted to a percentage, and then rules with absolute probabilities are added. If both a rule weight and a probability adjustment percentage are specified, the rule weight is first converted to a percentage, and then the value in the **Probability Adjust %** text box is added to determine the total weight for that rule.
4. Once you have modified the rule behavior to your requirements, click **Save** at the bottom of the page.

Related concepts

[Test Scores](#) (page 274)

[Creating New Anti-Spam Rules](#)

Note

It is recommended that you check with Sophos support before attempting to create any new anti-spam rules. If you are receiving false positives or false negatives, Sophos asks that you forward these messages to SophosLabs for our analysts to investigate. See "PureMessage Feedback" in the Contacting Sophos section for more information.

1. On the sidebar of the **Policy** tab, click **Anti-Spam Rules**.

The **Anti-Spam Options** page is displayed.

2. At the bottom of the page, click **New**.

A set of seven editable text boxes is displayed at the bottom of the page where you can set the information for the new anti-spam rule.

3. Fill in the information for the rule in each of the following text boxes:

- **Rule State:** Select **Auto**, **Enabled**, or **Disabled**.

Note

The **Auto** value sets the state of the rule according to whether there is a value in the rule's **Weight** or **Probability Adjust %** text box. If both scores are zero, the rule has no effect.

- **Rule Name:** The unique identifier for the rule. Using the default policy, rules matched by a message appear in a spam report header called the Rule Hit Rates report.
- **Desc:** A meaningful description for the rule.

- **Part:** The component of the message that is tested against the rule. The name of any message header can be specified; common headers include Subject, To and From. Specific message parts include:
 - **Envelope_To:** The recipient addresses, as interpreted from the SMTP "RCPT TO" command; the actual delivery address, as opposed to the message's To header.
 - **Envelope_From:** The sender's address, as interpreted from the SMTP "MAIL FROM" command.
 - **BODY:** Consecutive chunks of the message's body content (that is, paragraphs) as well as the message's Subject header; HTML parts are stripped of markup tags. Useful for matching words concealed by HTML tags.
 - **RAWBODY:** Consecutive chunks of the message's body content (that is, paragraphs) as well as the message's Subject header; markup tags in HTML parts are left intact. Useful for matching HTML markup characteristics.
 - **URI:** URI strings found in the body of the message.
 - **EOB:** The entire message body as well as the message's Subject header; HTML parts are stripped of markup tags. EOB is resource-intensive, as the entire message must be loaded at once. Use "BODY" if possible.
 - **RAWEOB:** The entire message body as well as the message's Subject header; markup tags in HTML parts are left intact. RAWEOB is resource-intensive because the entire message must be loaded at once. Use "RAWBODY" if possible.
 - **EOH:** All of the message's headers, concatenated into a single string.
 - **Full:** The entire message, including headers.
 - **Test:** The regular expression applied to the section of the message specified in the **Part** text box. The expressions must be enclosed in forward slashes ("/"). For example, to test for the occurrence of the word "opportunity", enter "/opportunity/" as the test. See the Regular Expression Primer in the Appendices for more information on regular expressions.
 - **Weight:** The value (or "weight") added to the message's total spam score when the message matches this rule. Values can be either positive or negative; prefix negative numbers with a minus symbol. For more information about how scores are calculated, see "Test Scores" in the Policy section of the *Administrator's Reference*.
 - **Probability Adjust %:** The absolute probability for the rule in the form of a percentage. When the total spam score is calculated for the message, rules with weights are first converted to a percentage, and then rules with absolute probabilities are added. If both a rule weight and a probability adjustment percentage are specified, the rule weight is first converted to a percentage, and then the value in the **Probability Adjust %** text box is added to determine the total weight for that rule.
4. Once you have set the information for the new rule, at the bottom of the page, click **Save**.

Related concepts

[Test Scores](#) (page 274)

Configuring Anti-Spam Options

You can set a variety of global anti-spam options and options for specific anti-spam feature groups.

To configure the anti-spam options:

1. On the sidebar of the **Policy** tab, click **Anti-Spam Options**.
The **Anti-Spam Options** page is displayed.
2. In the **Options** table, change any of the following text boxes as required:

- **Disable non-relay checks?:** Use the drop-down list to select **Yes** or **No**. The default setting is **No**.

When set to **Yes**, only the first received headers (written by the last server to relay the message) are subject to DNS checks. This provides both performance improvements and fewer false positives because trusted relays, as defined in the **Trusted Relay IPs** list, are exempt from certain DNS checks. Click **Edit trusted-relays** to edit the **Trusted Relay IPs** list.

- **Disable All Network Checks?:** Use the drop-down list to select **Yes** or **No**. The default setting is **No**.

When set to **Yes**, all types of network checks are disabled, including reverse DNS look-ups and other DNS checks. When set to **No**, individual network check rules can be enabled or disabled on the **Anti-Spam Rules** page.

- **Acceptable Character Sets:** Select the character sets from the list of those available by **Ctrl+clicking** the ones considered to be "safe".

Certain anti-spam rules are triggered based on the character set of the message. The selected "safe" character sets are exempted from this group of rules. The default character set is read from the system's LANG environment variable. The available character sets are:

- **Chinese, simplified and traditional:** (zh)
- **Cyrillic:** (ru, uk, tj, be, bg and ka)
- **Korean:** (ko)
- **Japanese:** (ja)
- **Thai:** (th)
- **English:** (en)

3. Once you have made the required changes in the **Options** table, click **Save**.
4. In the **Anti-Spam Feature Groups** table, set the feature groups and rules that you want enabled by doing the following:
 - a) Click the name of the feature group that you want to enable or disable.
The **Anti-Spam Options: <feature group>** page is displayed.
 - b) Use the **Enabled** drop-down list to select whether this feature group is **Enabled** or **Disabled**.
 - c) If you enabled the feature group, but want to specify which rules in the feature group are used, click **Rules** at the bottom of the **Properties** table.
The **Anti-Spam Rules** page for that feature group is displayed.
 - d) Click the name of a rule that you want **Enabled** or **Disabled**, set its usage from the drop-down list provided, and repeat for each rule that you want to enable or disable. Then click **Save** below the list of rules.
The **Anti-Spam Rules** page is redisplayed with the full list of rules.
5. Return to the **Anti-Spam Options** page and repeat steps "a." through "d." above for each feature group that you want to modify.
6. Click the **Restart now** button at the top of the page to restart the **Milter (Policy)** service.

2.3 Quarantine Tab

The **Quarantine** tab provides an interface for managing PureMessage quarantines, which are directories for temporarily storing problem or potential problem messages. Typically, quarantines are used for potential spam messages that can be reviewed later by administrators or end users to determine if they want to release (deliver) or delete them, or for virus-infected messages that you choose to retain for legal, organizational policy, or forensic reasons.

In addition, you can access many End User and Quarantine Digest options.

Quarantine Tab: Summary (Default) Page

The Quarantine: Summary page displays a graph that shows essential information about the PureMessage Quarantine. The graph shows:

- the maximum disk space available for the quarantine on the local system
- the size of the total quarantined messages in the current reporting period (two hours)
- the number of newly quarantined messages in the current reporting period (two hours)

Click on the graph to shift focus to the **Reports** tab, where you can change the report parameters.

Click View Quarantine Summary to return to this page from another Quarantine page.

Note

You can also generate, view, and export reports using the reporting options available via the **Reports** tab in the PureMessage Groups Web Interface. Although it is primarily used to delegate tasks under the group administration model, the Groups Web Interface can be configured as a reports manager. You may want to consider this alternative if your organization generates a lot of reports and you are comfortable managing PureMessage with multiple interfaces. See "Viewing and Managing Quarantine Search Results" in the "Administrative Groups" section of the *Administrator's Reference* for more information.

2.3.1 Managing the Quarantine

The **Quarantine > Manage Quarantine** page displays a set of query parameters used to search for messages in the quarantine. By default, the simple query form is displayed. Click **Advanced Mode** to access the additional search options.

Quarantined messages that match the query criteria are displayed in the results list, which shows one quarantined message per line. Click the envelope icon beside the list item to display the message content, quarantine information, and the message source.

Performing a Simple Quarantine Query

To perform a simple query of the quarantine:

1. Ensure that you are using the **Simple Mode** query on the **Manage Quarantine** page.
The **Simple Mode** provides only four query parameters.
2. Enter query parameters in one or more of the text boxes.
 - **Recipient:** The value entered is tested against the contents of the following message headers: `env_to`, `To`, `Cc`, `Bcc`, `Resent-To`, `Resent-Cc` and `Resent-Bcc`. (The `To` value in the message content is not tested, as this field is frequently forged.)

You can use the % symbol as a wildcard to match any set of characters within an address component. When using wildcards, the address's "@" symbol must be specified. For example, the following searches return `john.doe@example.com`:

```
%@example.com
john.%@example.com
%.doe@example.com
john.doe@example.%
%%
john.%%.com
john.doe%%
```

- **Sender:** The value entered is tested against the contents of the following message headers: `env_from`, `From`, `Sender`, `Reply-To`, `Resent-From` and `Resent-Sender`. (The `From` value in the message content is not tested because this field is frequently forged.)

You can use the % symbol as a wildcard to match any set of characters within an address component. When using wildcards, the address's "@" symbol must be specified. For example, the following searches return `john.doe@example.com`:

```
%@example.com
john.%@example.com
%.doe@example.com
john.doe@example.%
%%
john.%%.com
john.doe%%
```

- **ID:** The value entered is tested against the following message components:
 - **Queue ID:** Each message in the quarantine is assigned a unique Queue ID. To view a message's Queue ID, click the envelope icon beside the message, and then click **Quarantine Info**. To view a specific message, enter the message's Queue ID in this text box.
 - **Quarantine Digest ID:** When PureMessage generates quarantine digests, each message listed in the digest is assigned an ID code.
 - **Header Message ID:** If the format of the search string resembles a header message ID (for example, by containing an "@" separator), the contents of this text box are tested against the value in the `Message-ID` header. To view a message's `Message-ID` header, click the envelope icon beside the message and then click **Message Source**.

Wildcards are not supported in the **ID** text box.

- **Display:** Select the number of messages to display per page. The default is 20 messages.

3. Click **Run Query**.

Either a list of messages that match your query parameters is displayed, or a message is displayed advising that "No indexed messages matched this query." If there are no matches, change the parameters of your query and try again.

Performing a Advanced Quarantine Query

To perform an advanced query of the quarantine:

1. Ensure that you are using the **Advanced Mode** query on the **Manage Quarantine** page.
The **Advanced Mode** query contains over 15 query parameters.
2. Enter search criteria in the **Specify query parameters** form:

- **Recipient:** The value entered is tested against the contents of the following message headers: `env_to`, `To`, `Cc`, `Bcc`, `Resent-To`, `Resent-Cc` and `Resent-Bcc`. (The `To` value in the message content is not tested because this field is frequently forged.)

You can use the % symbol as a wildcard to match any set of characters within an address component. When using wildcards, the address's "@" symbol must be specified. For example, the following searches return `john.doe@example.com`:

```
%@example.com
john.%@example.com
%.doe@example.com
john.doe@example.%
%@%
john.%@%.com
john.doe@%
```

- **Sender:** The value entered is tested against the contents of the following message headers: `env_from`, `From`, `Sender`, `Reply-To`, `Resent-From` and `Resent-Sender`. (The `From` value in the message content is not tested, as this field is frequently forged.)

You can use the % symbol as a wildcard to match any set of characters within an address component. When using wildcards, the address's "@" symbol must be specified. For example, the following searches return `john.doe@example.com`:

```
%@example.com
john.%@example.com
%.doe@example.com
john.doe@example.%
%@%
john.%@%.com
john.doe@%
```

- **ID:** The value entered is tested against the following message components:
 - **Queue ID:** Each message in the quarantine is assigned a unique Queue ID. To view a message's Queue ID, click the envelope icon beside the message, and then click **Quarantine Info**. To view a specific message, enter the message's Queue ID in this text box.
 - **Quarantine Digest ID:** When PureMessage generates quarantine digests, each message listed in the digest is assigned an ID code.
 - **Header Message ID:** If the format of the search string resembles a header message ID (for example, by containing an "@" separator), the contents of this text box are tested against the value in the Message-ID header. To view a message's Message-ID header, click the envelope icon beside the message and then click **Message Source**.

Wildcards are not supported in the ID text box.

- **Subject:** The contents of this text box are tested against the subject of quarantined messages. If the search string is found anywhere within a message's subject line, the message will match.
- **Spam Rule:** The contents of this text box are tested against the names of all spam rules violated by the message. If the search string is found anywhere within any of the spam rules, the message will match.
- **Custom Reason:** The PureMessage policy can be configured to attach a reason to a message when it is quarantined. The default quarantine reasons can be specified in the **Reason** text box. If you have altered the default quarantine reasons in the policy, or added custom actions with non-default reasons, specify the custom reason in this text box. If the search string is found anywhere within a message's quarantine reason, the message will match.

- **Relay:** To select messages based on a specific server that passed the message to the internal server (the "relay"), enter the relay's hostname or IP address. If the search string is found anywhere within a message's relay, the message will match.
- **Milter Host:** If running multiple PureMessage servers and consolidating quarantined messages, select messages based on the PureMessage server that quarantined the message by entering the hostname in this text box. If the search string is found anywhere within a message's milter host, the message will match.
- **Maximum Age:** Use this text box to select messages that were quarantined within the specified number of hours. If a **Date Range** is also specified, the value in this text box is ignored.
- **Reason:** Select whether to display all messages, or only messages quarantined for a specific reason. With one exception, reasons included in the drop-down list correspond to reasons defined in versions of the default PureMessage policy. Although it is not used in the default policy, select the reason **Queue** to search for messages that have been approved but have yet to be delivered by pmx-queue. This applies to messages that have been approved by a user response to a Quarantine Digest, approved manually via the End User Web Interface, or approved through use of the Deliver immediately for action in the PureMessage policy script. Use the **Custom Reason** text box to specify reasons not included in the Reason drop-down list.

These reasons correspond to the reasons defined in the policy. Use the **Custom Reason** text box to specify a reason other than those available in this text box.

- **Order By:** Select the order in which the list is sorted. The **Probability** and **Probability (desc.)** sort options sort in ascending or descending order, according to the message's spam score. (Note that a message quarantined for a non-spam reason, such as a message containing a virus, could still have a spam score.)
- **Group By:** To collapse multiple messages with similar characteristics into a single line, select a grouping value from this text box. **Normalized Subject** strips the numbers and spaces from the beginning and end of the subject line, and group by the first fifteen characters; **Recipient** and **Sender** group by the sender and recipient stored in the message header (not the `env_from` and `env_to` values).
- **Spam Probability:** To select messages within a range of spam probabilities, enter the range in these text boxes.
- **Display:** Select the number of messages to display per page. The default is 20 messages.
- **Date Range:** Use these text boxes to select messages based on the date they were quarantined. If the **Date Range** check box is selected, any value selected in the **Maximum Age** text box is ignored.

3. Click **Run Query**.

Either a list of messages that match your query parameters is displayed, or a message is displayed indicating that "No indexed messages matched this query." If there are no matches to your query, change the parameters of your query and try again.

Viewing Quarantined Message Details

Once you have performed a successful simple or advanced query, you can view additional details as described below.

- *To view information about a quarantined message:* Click the envelope icon on the left side of a quarantined message row.

A **Message Preview** dialog box is displayed, with the **Content** tab showing the **From**, **To** and **Subject** fields, as well as the body of the message.

- *To see the quarantine information for the message:* Click on the **Quarantine Info** tab. The following details are displayed: A summary of the message, the spam rules that were violated

by the message, the envelope from and to values, the hosts that handled the message, and the message's queue ID.

- *To see the raw message, including full headers:* Click on the **Message Source** tab.

Managing Quarantined Messages

To perform management operations on quarantined messages:

1. Select the check box to the left of the message(s) on which you want to perform a particular management action.
2. Click the button for the action that you want to perform:
 - **Approve:** Approves quarantined messages for delivery to the original recipient. A copy of the approved message is sent to the original recipient, and a copy is kept in the quarantine (in the `var/qdir/sent` directory). These are eventually deleted (or archived) by the `pmx-qexpire` scheduled service. If you attempt to approve a message containing a virus, PureMessage prompts you to confirm the approval. A message appears at the top of the query results page indicating the number of messages approved.
 - **Delete:** Moves selected messages to the `var/qdir/trash` directory. These messages are eventually deleted (or archived) via the `pmx-qmeta-index` scheduled service.
 - **Forward:** Forwards one or more messages to a specified recipient. If you select this management option, a **Forward** message form is displayed. Enter the email address to which you want the message forwarded, and click **Send**. The message is not removed from the quarantine queue.
 - **Save:** Saves selected messages retrieved from the quarantine. Saved messages are exported to a file in mbox format. PureMessage prompts you to enter a destination file for the message(s). If you select this management option, a **Save** message form is displayed. Enter the destination filename and click **Save**. Saved messages are not removed from the quarantine.

Alternatively, you can click **Delete All**, which sets all the messages that matched your query to be moved to the `var/qdir/trash` directory the next time that `pmx-qmeta-index` is run.

2.3.2 Setting End User Options

You can, optionally, grant end users access to the End User Web Interface (EUWI), a web page with limited access to email-filtering features. For example, you may want to allow them to review their quarantined messages to see if any messages have been incorrectly identified as spam (false positives). The **End User Options** page allows you to set which options will be available to end users.

To set the basic end user options:

1. In the **Enduser** section of the **Quarantine** tab's sidebar, click **End User Options**.
The **End User Options** page is displayed.
2. Set the end user options:
 - **Enduser URL:** The URL location of the EUWI. Can be configured to run the interface outside of the current PureMessage server.
 - **User List:** Specify the list of users authorized to access the EUWI. By default, this is set to the "End Users" list; by default, this list contains "***@**", which authorizes all users to access the interface. To alter the default list, click **End Users** on the sidebar of the **Policy** tab.
 - **Quarantine Reasons:** When messages are quarantined, the policy script assigns a reason that describes why the message was quarantined (for example, "spam"). The messages displayed in the EUWI are determined by the reasons entered in this text box. For example, if "spam" is

specified, (the default), only messages quarantined with the reason "spam" will be displayed. This text box is not case sensitive, but it only accepts alphanumeric characters.

In the following text boxes, use abbreviation suffixes to specify the unit of time: "s" (seconds), "m" (minutes), "h" (hours) and "w" (weeks). For example, to specify twenty minutes, enter "20m"; to specify an hour and a half, enter "1h30m".

- **Session Expire:** Sets the amount of time before the end user's session cookie expires. When the cookie expires, the user must request another password. This text box has no effect when flat-file authentication is in use.
- **Query Delay:** Sets the amount of time allowed between each query made to the PureMessage server from the EUWI. This can be used to limit the rate of queries by end users.
- **GMT Offset:** In the quarantine, message dates and times are stored in GMT (Greenwich Mean Time). The value displayed in this text box will be added to (or subtracted from) message's date and time so that the message is displayed in "local" time. For example, if the timestamp on a message is 9 AM GMT, and the value in this text box is set to "2h", the message time displayed in the EUWI is 11 AM. Precede the entry by a minus symbol to subtract from GMT.

Although this value is shown in the Manager interface, it must be configured at the command line. See "Adjusting the GMT Offset for the End User Web Interface" in the *Administrator's Reference* for more information.

3. Click **Save**.

A message is displayed, warning you that the HTTPD (RPC/UI) service, which is the web service for the end user access, needs to be restarted.

4. Click **Restart now**.

A message is displayed, informing you that the HTTPD (RPC/UI) service has been sent a restart signal.

Related concepts

[End User Web Interface](#) (page 72)

Related tasks

[Adjusting the GMT Offset for the End User Web Interface](#) (page 296)

2.3.3 Configuring End User Features

If you have granted end users access to the End User Web Interface (EUWI), the **Configure End User Features** page allows you to set which features are available.

To configure the features available to end users:

1. In the **Enduser** section of the **Quarantine** tab's sidebar, click **Configure End User Features**.

The **Configure End User Features** page is displayed.

2. Select the check boxes beside any of the following options to grant end users access to these features:

- **View Quarantined Messages:** Provides end users with the 'Blocked Messages' interface. Users can view all quarantined messages sent to them.
- **View Message Contents:** Provides end users with the ability to click a message's **Subject** header and read the quarantined message. This functionality is available via the PureMessage quarantine or the EUWI's "Blocked Messages" page.
- **User White List:** End users can specify individual whitelisted senders. This functionality is called "Approved Senders" in the EUWI.

- **User Black List:** End users can specify individual blacklisted senders. This functionality is called “Blocked Senders” in the EUWI.
 - **White list and Black list Wildcards:** End users can specify individual whitelisted and blacklisted senders or hosts using wildcards. This functionality is only available to end users who select a sender to allow or block.
 - **Anti-Spam Opt-Out:** An end user mail filtering preference. Users can disable all spam and offensive content blocking.
 - **Digest Opt-Out:** Users can opt out of message digests.
 - **Vacation Hold:** The amount of time that messages are held in the quarantine, as determined by the setting in `/opt/pmx6/etc/pmx.d/quarantine_expire.conf`.
 - **Notification Email:** Specify whether the EUWI includes a link for users to request their password. If this option is enabled, users can request that their account information be sent. Be sure to enable this option if **SessionID is emailed to user** is the method you have selected on the **End User Authentication** page.
 - **Language Preference:** Determines whether end users can adjust the language setting via the EUWI. When this check box is selected, end users have access to the Language Preference drop-down list, from which they can select their desired language. This setting overrides the language specified in the **Default Language** (described below) drop-down list. When the check box is not selected, the **Language Preference** drop-down list is not available in the EUWI.
 - **Max Hold Duration:** The maximum time period (as measured from the current date) that end users can specify that messages be held. The default is one week. The overall length of the hold period will vary, based on this setting and the quarantine expiry period that is set in `/opt/pmx6/etc/pmx.d/quarantine_expire.conf`.
 - **Max Items Per List:** The maximum number of entries that end users are allowed to add to their whitelists and blacklists.
 - **Default Language:** Changes the default language preference set during the PureMessage installation. If the **Language Preference** (described above) check box is selected, individual user preferences specified via the **Language Preference** drop-down list in the EUWI will override this setting.
3. Click **Save**.
- A message is displayed, warning you that both the Milter (Policy) service and the HTTPD (RPC/UI) service need to be restarted.
4. Click **Restart now** beside each message.
- A message is displayed after you press each button, informing you that those services have been sent restart signals.

Related concepts

[End User Web Interface](#) (page 72)

2.3.4 Configuring End User Authentication

If you have granted end users access to the End User Web Interface (EUWI) to manage certain aspects of email-filtering, you must configure the method they will use to authenticate their identities when they log in. The **Quarantine: End User Authentication** page allows you to set the authentication method that is used to check their identity.

To set the end user authentication method:

Note

You can also configure PureMessage to use multiple forms of authentication. For example, your organization may use more than one type of LDAP to authenticate users. For instructions, see [Configuring Multiple Authenticators](#) in the Sophos Knowledgebase.

1. In the **Enduser** section of the **Quarantine** tab's sidebar, click **End User Authentication**.
The **End User Authentication** page is displayed.
2. Select the option button for the end user authentication method that you want to use:
 - **SessionID is emailed to user:** This option generates a session ID that is emailed to the end user and is valid for a specified length of time. For this method, you must set the following options:
 - **Email template:** Specify the path and filename of the email template for the generated session ID that becomes the user's password.
 - **Session expiry time:** Specify the length of the end user login session.

Note

Use abbreviation suffixes to specify the unit of time: "s" (seconds), "m" (minutes), "h" (hours) and "w" (weeks). For example, to specify 20 minutes, enter "20m"; to specify an hour and a half, enter "1h30m".

- **Password database is kept in a plain text file:** This option allows you to use a plain text password file consisting of a comma-separated list of one username and one password per line. This file may be encrypted. For this method, you must set the following options:
 - **File path:** Specify the path and filename of the password text file. The default is `enduser/enduser_ui_user_passwords`.
 - **Encryption:** Specify the encryption applied to the file. The recognized options are `none`, `crypt` or `md5`.
- **LDAP based authentication:** This option allows you to use an existing LDAP server as the source for your end user lists and authentication. For this method, you must set the following options:
 - **LDAP Server:** Specify the 'host:port' of the server(s) to connect to when authenticating users via LDAP. To specify more than one LDAP host for failover (which is strongly advised), enter a semicolon-separated list of hosts. If no port is specified, port 389 is used by default.

To use an encrypted LDAPS connection, simply prefix the host:port with `ldaps://`. For LDAPS connections, port 636 is used if no port is specified. For example:

`localhost:389 ldaps://ldap.mycompany.com`
 - **DN for binding to LDAP server:** Specify the Distinguished Name (DN) used to connect to the LDAP server in order to query the Distinguished Name of the user the system is attempting to authenticate. This text box supports variable substitution (described below).
 - **Password for binding to LDAP server:** Specify the password used to connect to the LDAP server in order to query the Distinguished Name (DN) of the user that the system is attempting to authenticate. This DN and password should be granted minimal rights but must be able to perform a query to retrieve the DN for a user based on their provided username and ID.

- **Base DN for user accounts:** Specify the top LDAP directory node from which the search is performed to retrieve the DN of the user to be authenticated. This text box supports variable substitution (described below).
- **Filter to find user account:** Specify the LDAP query that is performed to retrieve the DN of the user account to be authenticated. This filter should only return a single result record. You may experience inconsistent behavior if the filter returns multiple results. This text box supports variable substitution (described on the `ldap.conf` man page).

Note

For LDAP, the configuration setting for the user account should be `(sAMAccountName=%%username%%)`, not `(uid=%%username%%)`.

Variable substitution can be used in the **DN for binding to LDAP server**, **Base DN for user accounts** and **Filter to find user account** LDAP text boxes. Variable substitution permits the insertion of information as variables using a pre-defined syntax. The following variables are available for substitution:

- `%%username%%` - The full username as provided by the user on the login page.
- `%%password%%` - The full password as provided by the user on the login page.
- `%%bind_dn%%` - The 'Bind DN' as specified in the configuration.
- `%%base_dn%%` - The 'Base DN' as specified in the configuration.

Note

Advanced LDAP configuration options are available via `ldap.conf`, located by default in `/pmx/etc/enduser/auth.d`. Advanced options are described on the `ldap.conf` man page. These options are recommended for use by advanced administrators only.

3. Click **Save**.

Testing Authentication

You can test any of the three authentication methods: SessionID, plain text file or LDAP-based authentication.

To test authentication:

1. On the **End User Authentication** page, type an existing and known username and password in the appropriate text boxes.
If you are testing LDAP authentication, the user's email address must be present in the user's LDAP profile or the test will fail.
2. Click **Test**.

Related concepts

[End User Web Interface](#) (page 72)

Related information

[ldap.conf](#)

About End User Authorization Methods

The authentication methods by which users access the End User Web Interface (EUWI) are set using the End User Authentication feature on the **Quarantine** tab of the PureMessage Manager. The default authentication method is to email a session ID to the end user. The alternate methods are to authenticate through an encrypted password file or using LDAP. All three methods are described below.

SessionID Authentication

This default end user authentication method is based on emailing a generated session ID key to the end user. The session ID key is invalid after the **Session expiry time**, which is defined using abbreviation suffixes to specify the unit of time: "s" (seconds), "m" (minutes), "h" (hours) and "w" (weeks). So, two days, three hours and twenty minutes would be entered as 51h20m. The email sent to the end user is based on an **Email template**, which can be modified if required. (We suggest that you make any modifications to a copy or back up the original.)

When end users first access the web interface URL, (<EUWI_host>.<domain>28080), they are prompted to enter their email address and request a password. The generated session ID key is sent to the specified email address as their password. If the user requests a password multiple times, only the most recently generated password is valid. After receiving a password, end users can log in to the EUWI.

Password Text File Authentication

End user authentication can also be configured to use a text passwords flat-file database. To configure this usage, change the `auth` variable in the `enduser.conf` configuration file (located by default in the `/opt/pmx6/etc/enduser` directory) to `flat_file`. Next, edit the `enduser_ui_user_passwords` file, and add the desired usernames and passwords using the commented examples in the file as a model. Restart the HTTP (RPC/UI) service to make the changes active. After re-starting the EUWI, login authentication is controlled according to the username/password combinations in `enduser_ui_user_passwords`, so these passwords must be emailed to the end users.

There are three methods of storing each user's password: plain text (the default), crypt, and md5. To configure the password storage format, add the usernames and passwords to the `enduser_ui_user_passwords` file. Then, in the `etc/enduser/auth.conf` file, set the "crypt" option in the `<Authenticator flat_file> -> <config>` section to the desired method.

This is a slightly simpler process for end users because they do not need to request a password. It does require more work by the PureMessage administrator, as the end users' assigned passwords must be emailed to them along with the URL to access the EUWI.

LDAP-Based Authentication

End user authentication can also be configured to use an existing LDAP directory, such as Active Directory, Sun ONE Directory Server 5.2, and OpenLDAP. For more about configuring end user authentication, see the PureMessage Manager Reference. In general, specify the "host:port" of the LDAP server(s), the LDAP server's Distinguished Name(DN), a password to access LDAP server information, if required, the base DN for user accounts, and the filter translation of the field name for the LDAP data that you are querying.

Once LDAP authentication is configured, you must enable the End User Web Interface for LDAP. In the `etc/enduser/enduser.conf` file, edit the `auth` option so that it reads `auth=ldap`. Restart the EUWI to make this change take effect.

Depending on whether you are authenticating users by email address or Active Directory ID, you may want to edit the login page for the EUWI so that it displays an appropriate message to your users. The template for this page can be found in `lib/manager/HTTPD/tmpl/authorize.html`.

All errors and warning messages returned from an LDAP server are placed in the `var/log/manager/httpd_error.log` file. All items related to LDAP Authentication are prefixed by the phrase "EU-LDAP-AUTH", making it easier to separate them from other entries in the log file.

Related concepts

[End User Web Interface](#) (page 72)

Related information

[Accessing the End User Web Interface](#)

[ldap.conf](#)

2.3.5 Managing End User Lists

Whitelists are used to allow trusted email addresses and hosts to bypass normal spam filtering, which ensures that messages from the listed sources are delivered. Whitelists can be maintained by individual users, enabling personalized processing. End users can be granted the ability to manage their own whitelists through the End User Web Interface (EUWI) or, if access is not granted, administrators can maintain end users' personal whitelists.

Blacklists are used to disallow problem email addresses and hosts (for example, those known to be senders of viruses or spam) before normal spam filtering, which ensures that messages from the listed sources will never be delivered. Blacklists can be maintained by individual users, enabling personalized processing. End users can be granted the ability to manage their own blacklists through the End User Web Interface (EUWI) or, if access is not granted, administrators can maintain end users' personal blacklists.

Related concepts

[End User Web Interface](#) (page 72)

Adding an End User Whitelist

1. In the **Enduser** section of the **Quarantine** tab's sidebar, click **Edit End User Whitelist**.

The **Edit List: whitelisted-senders-per-user** page is displayed.

2. In the **Key** text box, enter the end user's email address for whom you want to add a whitelist, and then click **Add**.

The **Edit List: whitelisted-senders-per-user: <endUser@email.address>** page is displayed.

3. Perform one or more of the following operations:

- **Add List Items:** In the **Add Items** text box, add all trusted sender email addresses and hosts that you want to add to the end user's whitelist, and then click **Add**.

All added whitelisted senders appear in the **List Items** menu.

- **Edit a List Item:** In the **List Items** list, select the check box beside the trusted email address or host that you want to edit, and click **Edit**.

The selected entry is displayed in an editable text field.

Make the desired changes, and click **Save**.

The list is redisplayed with the changed entry.

- **Delete a List Item:** In the **List Items** list, select the check box beside the trusted email address or host that you want to delete, and click **Delete**.

The list is redisplayed without the deleted entry.

Note

When adding an end-user whitelist you must populate that whitelist with at least one trusted sender or host for the list to be created. Also, adding an end-user whitelist does not automatically add a blacklist for that user; that must be done separately.

Editing an End User Whitelist

After end users' whitelists have been created, you can modify the contents of those lists.

To edit an end user's whitelist:

1. In the **Enduser** section of the **Quarantine** tab's sidebar, click **Edit End User Whitelist**.

The **Edit List: whitelisted-senders-per-user** page is displayed.

2. In the **Available List Keys** list, click the email address of the user whose whitelist you want to modify.

The **Edit List: whitelisted-senders-per-user: <endUser@email.address>** page is displayed.

3. Perform one or more of the following operations:

- **Add List Items:** In the **Add Items** text box, add all trusted sender email addresses and hosts that you want to add to the end user's whitelist, and then click **Add**.

All added whitelisted senders appear in the **List Items** menu.

- **Edit a List Item:** In the **List Items** list, select the check box beside the trusted email address or host that you want to edit, and click **Edit**.

The selected entry is displayed in an editable text field.

Make the changes that you want and click **Save**.

The list is redisplayed with the changed entry.

- **Delete a List Item:** In the **List Items** list, select the check box beside the trusted email address or host that you want to delete, and click **Delete**.

The list is redisplayed without the deleted entry.

Deleting an End User Whitelist

After end users' whitelists have been created, you can delete any one of those lists.

To delete an end user's whitelist:

1. In the **Enduser** section of the **Quarantine** tab's sidebar, click **Edit End User Whitelist**.

The **Edit List: whitelisted-senders-per-user** page is displayed.

2. In the **Available List Keys** list, select the check box beside the email address of the user whose whitelist you want to delete, and click **Delete**.

The **Available List Keys** list is redisplayed with the selected user's whitelist removed from the list.

Adding an End User Blacklist

1. In the **Enduser** section of the **Quarantine** tab's sidebar, click **Edit End User Blacklist**.

The **Edit List: blacklisted-senders-per-user** page is displayed.

2. In the **Key** text box, enter the end user's email address for which you want to add a blacklist, and then click **Add**.

The **Edit List: blacklisted-senders-per-user: <endUser@email.address>** page is displayed.

3. Perform one or more of the following operations:

- **Add List Items:** In the **Add Items** text box, add all problem sender email addresses and hosts that you want to add to the end user's blacklist, and then click **Add**.

All added blacklisted senders appear in the **List Items** menu.

- **Edit a List Item:** In the **List Items** list, select the check box beside the problem email address or host that you want to edit, and click **Edit**.

The selected entry is displayed in an editable text field.

Make the changes that you want and click **Save**.

The list is redisplayed with the changed entry.

- **Delete a List Item:** In the **List Items** list, select the check box beside the problem email address or host that you want to delete, and click **Delete**.

The list is redisplayed without the deleted entry.

Note

When adding an end-user blacklist you must populate that blacklist with at least one problem sender or host for the list to be created. Also, adding an end-user blacklist does not automatically add a whitelist for that user; that must be done separately.

Editing an End User Blacklist

After end users' blacklists have been created, you can modify the contents of those lists.

To edit an end user blacklist:

1. In the **Enduser** section of the **Quarantine** tab's sidebar, click **Edit End User Blacklist**.

The **Edit List: blacklisted-senders-per-user** page is displayed.

2. In the **Available List Keys** list, click the email address of the user whose blacklist you want to modify.

The **Edit List: blacklisted-senders-per-user: <endUser@email.address>** page is displayed.

3. Perform one or more of the following operations:

- **Add List Items:** In the **Add Items** text box, add all problem sender email addresses and hosts that you want to add to the end user's blacklist, and then click **Add**.

All added blacklisted senders appear in the **List Items** menu.

- **Edit a List Item:** In the **List Items** list, select the check box beside the problem email address or host that you want to edit, and click **Edit**.

The selected entry is displayed in an editable text field.

Make the changes that you want and click **Save**.

The list is redisplayed with the changed entry.

- **Delete a List Item:** In the **List Items** list, select the check box beside the problem email address or host that you want to delete, and click **Delete**.

The list is redisplayed without the deleted entry.

Deleting an End User Blacklist

After end users' blacklists have been created, you can delete any one of those lists.

To delete an end user blacklist:

1. In the **Enduser** section of the **Quarantine** tab's sidebar, click **Edit End User Blacklist**.

The **Edit List: blacklisted-senders-per-user** page is displayed.

2. In the **Available List Keys** list, select the check box beside the email address of the user whose blacklist you want to delete, and click **Delete**.

The **Available List Keys** list is redisplayed with the selected user's blacklist removed from the list.

2.3.6 Managing Quarantine Digest Rules

Quarantine digests are lists of quarantined messages that are sent to end users. These digests are sent to the end users included in the **Quarantine Digest Users** list. A user can release messages from the quarantine by replying to the digest and deleting all lines except those that represent messages the user wants to receive.

Adding Quarantine Digest Rules

1. In the **Quarantine Digests** section of the **Quarantine** tab's sidebar, click **Digest Rules**.

The **Digest Rules** page is displayed.

2. Click **Add** below the **Digest configurations** table.

The **Add digest** form is displayed.

3. Set the digest rule information:

- **Name:** Enter a descriptive name for the digest.
- **Template:** Enter the name of the file that contains the template for the digest. This file must be stored in the `/opt/pmx6/etc/templates/<langdir>` directory.
- **Session Expire:** Sets the amount of time before the end user's session cookie expires. When the cookie expires, the user must request another password. This field has no effect when flat-file or LDAP authentication is in use.
- **Reasons:** When a message is quarantined, the quarantine record contains the "reason" why the message was not delivered, such as "spam" or "virus". Digests are generated for messages that match the reason specified in this field.
- **Address List:** Select from the drop-down list a list of email addresses which should receive digests. For information on creating these address lists, see the help on the **Quarantine Digest Users** page.

4. Click **Save**.

The **Digest configurations** table is displayed, showing your changes, and a message is displayed, informing you of the result of the save operation.

Editing Quarantine Digest Rules

1. In the **Quarantine Digests** section of the **Quarantine** tab's sidebar, click **Digest Rules**.
The **Quarantine : Digest Rules** page is displayed.
2. In the **Digest configurations** table, click the name of the digest configuration that you want to edit.
The **Edit digest** form is displayed.
3. Make the changes that you want to the digest rule information:
 - **Name:** Enter a descriptive name for the digest.
 - **Template:** Enter the name of the file that contains the template for the digest. This file must be stored in the `/opt/pmx6/etc/templates/<langdir>` directory.
 - **Session Expire:** Sets the amount of time before the end user's session cookie expires. When the cookie expires, the user must request another password. This text box has no effect when flat-file or LDAP authentication is in use.
 - **Reasons:** When a message is quarantined, the quarantine record contains the "reason" why the message was not delivered, such as "spam" or "virus". Digests are generated for messages that match the reason specified here. This text box accepts alphanumeric characters only.
 - **Address List:** Select from the drop-down list a list of email addresses which should receive digests. For information on creating these address lists, see the help on the **Quarantine Digest Users** page.
4. Click **Save** to implement your changes.
The **Digest configurations** table is displayed, showing your changes, and a message is displayed, informing you of the result of the save operation.

Deleting Quarantine Digest Rules

1. In the **Quarantine Digests** section of the **Quarantine** tab's sidebar, click **Digest Rules**.
The **Digest Rules** page is displayed.
2. In the **Digest configurations** table, select the check box beside the name of the digest configuration that you want to delete, and click **Delete**.
The **Digest configurations** table is redisplayed, with the selected digest configuration rule removed from the list.

Consolidating the Default Digest Types (Spam, Offensive)

By default, PureMessage generates two types of digests for quarantined messages: "spam" and "offensive". The "spam" digest reports messages quarantined for containing spam with the reason "spam". The "offensive" digest reports messages quarantined for containing offensive words with the reason "offensive". In some cases, users may prefer to receive a single digest containing both quarantine reason types.

To combine both the default offensive words and spam digests into a single digest:

1. On the **Quarantine** tab, click **Digest Rules**.
The **Digest Rules** page is displayed.
 2. Add the reason "offensive" to the spam digest.
 3. Delete the digest called "offensive". (See "Deleting Quarantine Digest Rules" for instructions.)
- The spam digest now includes both types of messages.

2.3.7 Setting Quarantine Digest Options

If you decide to use quarantine digests, you can set the approval address alias and the expiry period on the **Digest Options** page. The approval address alias is the reply-to email address that is set in the quarantine digests that are emailed to end users, and it is this email address to which your end users respond to release or delete their messages from the quarantine. The expiry period is the length of time that end users have to respond to the quarantine digests.

To set the quarantine digest options:

1. In the **Quarantine Digests** section of the **Quarantine** tab's sidebar, click **Digest Options**.

The **Digest Options** page is displayed.

2. Change the default digest options as required:

- **Approve Address:** During the PureMessage installation, an email alias is added for message approvals. By default, this alias is set to call `pmx-auto-approve`. (If using a sendmail or Postfix distribution other than the one included with PureMessage, you must manually create the alias. See Quarantine Digests in the PureMessage User Guide for instructions.) The address specified in this field must match the sendmail or Postfix alias.
- **Expiry Period:** PureMessage stores a record of each quarantine digest. When an end-user requests the release of a message, the request is validated against the archived digest records. Periodically, (by default, every 5 days) the digest archive should be cleared.

3. Click **Save**.

The **Digest Options** page is redisplayed, and a message is displayed, informing you of the result of the save operation.

2.3.8 Managing the Quarantine Digest Users List

To send out quarantine digests to end users, you must build a quarantine digest users list, which is a simple list of email addresses for your end users. You can then edit and delete list entries as necessary.

Adding Users to a Quarantine Digest Users List

1. In the **Quarantine Digests** section of the **Quarantine** tab's sidebar, click **Quarantine digest users**.

The **Edit List: Quarantine digest users** page is displayed.

2. In the **Add Items** text box, enter one email address per user for the end users that you want to add to the quarantine digest user list, and then click **Add**.

To specify certain addresses or types of addresses as non-recipients, enter an exclamation point at the beginning of the address (for example, `!example@foo.com`). To exclude certain types of addresses, enter a partial address that acts as a wildcard (for example, `!postmaster@`).

The **List Items** list is redisplayed with the added users included.

Editing a Quarantine Digest Users List Entry

1. In the **Quarantine Digests** section of the **Quarantine** tab's sidebar, click **Quarantine digest users**.

The **Edit List: Quarantine digest users** page is displayed.

2. In the **List Items** table, select the check box beside the email address of the end user that you want to edit, and then click **Edit**.

The selected email address in the **List Items** table is displayed in an editable text box.

3. Edit the end user's email address as required, and then click **Save**.

The **List Items** table is redisplayed, with the email address changes.

Deleting an Entry from the Quarantine Digest Users List

After end users' whitelists have been created, you can delete any one of those lists.

To delete an end user's whitelist:

1. In the **Quarantine Digests** section of the **Quarantine** tab's sidebar, click **Quarantine digest users**.

The **Edit List: Quarantine digest users** page is displayed.

2. In the **List Items** table, select the check box beside the email address of the end user that you want to delete, and then click **Delete**.

The **List Items** table is redisplayed, with the selected email address removed from the list.

Creating Customized Quarantine Digest Users Lists

Creating multiple **Quarantine Digest Users** lists allows you to send out quarantine digests to lists of recipients in different time zones, at different times, or to send out digests of messages quarantined for different reasons to different groups of end users.

To create a customized digest user list:

1. On the **Policy** tab in the PureMessage Manager, and click **New** at the top of the **Lists** sub-menu.

The **Policy: Add List/Map** page is displayed.

2. Fill in the form as follows:

- a) Select **List** as the **Type**.
- b) For both the **ID** and the **Name**, enter a name that begins with `digest-` and is descriptive of the list's purpose, such as `digest-secondshift`.
- c) For the **Match Type**, select **Email Globs**.
- d) Select the **Source** option of your choice, and fill in any information as required.
- e) Click **Save**.

You are prompted to add items to the newly created list.

- f) Add the email addresses of the "secondshift" digest users to this list. See **Match Types** in the Policy Tab section for more information.
3. Remove the email addresses of the "secondshift" digest users from the default Quarantine Digest Users list by clicking **Quarantine digest users**, selecting the check boxes beside the users that are in the "secondshift" quarantine digest users list, and clicking **Delete**.
 4. Add this new Digest Users list to the Digest Rules by clicking **Digest Rules** on the sidebar of the **Quarantine** tab, and clicking **+Add** below the **Digest configurations**.

The **Add digest** form is displayed.

5. Fill in the form as follows:

- a) **Name:** Enter a descriptive name, like "secondshift".
- b) **Template:** You can use the existing spam digest template `/opt/pmx6/etc/digest-spam.tmpl`.
- c) **Reasons:** Enter the appropriate reason, for example, `spam`.

- d) **Address List:** From the drop-down list, select the address list that you created in step 2.
- e) Click **Save**.

You are returned to the **Digest configurations** table with the new digest rule appearing in the table.

6. Set the Scheduler to create the desired quarantine digests by adding another **pmx-qdigest** scheduled job that uses this command:

```
/opt/pmx64/bin/pmx-qdigest --digest=secondshift --quiet
```

And set the schedule for the job as appropriate for the new quarantine digest users list.

Related concepts

[Match Types](#) (page 115)

Related information

[pmx-qdigest.conf](#)

[lists.conf](#)

[maps.conf](#)

[pmx-qdigest-init](#)

2.3.9 Managing the Notifications Address Map

The Notifications Address Map allows you to redirect digests that are bound for a specific account to a different account.

You can add mapping entries to the Notifications Address Map, and you can edit and delete them as necessary.

Adding Redirects to the Notifications Address Map

1. In the **Quarantine Digests** section of the **Quarantine** tab's sidebar, click **Notifications address map**.
The **Edit Map: Notifications address map** page is displayed.
2. In the **Map From** text box, enter the email address that the quarantine digest is sent to.
3. In the **Map To** text box, enter the email address that you want the quarantine digest redirected to, and click **Add**.

The **Edit Map: Notifications address map** page is redisplayed, with the added map included.

Editing Redirects in the Notifications Address Map

1. In the **Quarantine Digests** section of the **Quarantine** tab's sidebar, click **Notifications address map**.
The **Edit Map: Notifications address map** page is displayed.
2. In the list of redirection mappings, select the check box beside the mapping that you want to edit, and click **Edit**.

The **Edit Map: Notifications address map** page is redisplayed, with the selected redirection mapping loaded into the **Map From** and **Map To** fields.

3. Make the required changes, and click **Save**.

The **Edit Map: Notifications address map** page is redisplayed with the changes included.

Deleting Redirects from the Notifications Address Map

1. In the **Quarantine Digests** section of the **Quarantine** tab's sidebar, click **Notifications address map**.

The **Edit Map: Notifications address map** page is displayed.

2. In the list of redirection mappings, select the check box beside the mapping that you want to delete and click **Delete**.

The **Edit Map: Notifications address map** page is redisplayed, with the selected map removed.

2.4 Delayed Mails Tab

The **Delayed Mails** tab displays a set of query parameters used to search for Delayed messages. Delayed messages are suspect emails that are held for a specified period of time. By delaying delivery, the appliance creates a timed interval during which new anti-spam definitions created by Sophos Labs can be received, after which the appliance can rescan the messages using the most up to date information possible.

Enter search criteria in the **Specify query parameters** form:

- **Recipient:** Enter a recipient's email address.
- **Sender:** Enter a sender's email address.
- **Milter Host:** If running multiple Edge servers, enter the name of a host for which you want to view the delayed messages.
- **Mails delayed up to:** Use this text box to select messages that were delayed within the specified number of minutes.
- **Display:** Select the number of messages to display per page.

Click **Run Query**.

Either a list of messages that match your query parameters is displayed, or a message is displayed indicating that "No indexed messages matched this query." If there are no matches to your query, change the parameters of your query and try again.

2.4.1 Viewing Delayed Message Details

Once you have performed a successful simple query, you can view additional details as described below.

- **From:** Displays the sender's email address.
- **To:** Displays the recipient's email address.
- **Date:** The date and time at which the message was delayed.
- **Size:** Displays the size of the message (KB).
- **Will be re-scanned at:** The time at which the messages categorized as suspicious, will be rescanned. The current time is in 24-hour format.

2.4.2 Managing Delayed Messages

To perform management operations on delayed messages:

1. Select the check box to the left of the message(s) on which you want to perform a particular management action.
2. Click the button for the action that you want to perform:
 - **Re-inject:** Re-inject the selected messages in to the militer NOW (normally, they are re-injected after a stipulated delay period).
 - **Delete:** The selected message will be permanently removed from the edge servers.
 - **Delete All:** All messages will be permanently removed form the edge servers.

2.5 Reports Tab

The Reports tab allows you to generate and view a variety of reports on PureMessage performance, operations, and message-processing statistics that are displayed as graphs and tables. Reports can be exported to CSV format, scheduled for automatic generation, and emailed to specified recipients.

To use these reporting features, the PostgreSQL database must be enabled. Several programs are used to populate some database tables prior to initial use. Others consume report data from PureMessage log files and store the data in the PostgreSQL database.

Note

If the database connection becomes unavailable during normal operation, only reports depending on OS health data and quarantine data are affected. Once the database connection is re-established, all other reports will reflect up-to-date system information.

The **Reports: Summary** (default) page displays three of the key PureMessage reports:

- **Top Spam Relays:** The top relays by number of spam messages received from them.
- **Message Categorizations:** The number of messages detected as spam, virus or other (neither virus nor spam). If PureMessage determines that a message contains spam and also contains a virus, the message counts toward the virus total only. A message is marked as being spam if its spam probability score is 50% or greater by default.
- **Quarantine Size:** The current size (in kilobytes) of local and central quarantines. It also summarizes new messages added, helping you understand the growth rate of your quarantine.

To update the reports on the **Summary** page, click **Reports Summary** on the sidebar. This link is also used to return to the **Summary** page from other pages of the **Reports** tab.

Tip

Alternatively, you can generate, view, and export reports using the reporting options available via the **Reports** tab in the PureMessage Groups Web Interface. You can also use the Groups Web Interface to create custom policy reports. Although it is primarily used to delegate tasks under the group administration model, the Groups Web Interface can be configured as a reports manager. You may want to consider this alternative if your organization generates a lot of reports and you are comfortable managing PureMessage with multiple interfaces. See "Viewing and Managing Reports" in the "Administrative Groups" section of the *Administrator's Reference* for more information.

Related concepts

[Logs and Reports](#) (page 298)

This section describes the management of the log files that register activities and the activities that are logged. It also covers generation of reports that are drawn, in part, from this data.

[Administrative Groups](#) (page 201)

This section describes the setup and Management of the Groups Web Interface, which a system administrator can use to delegate selected tasks to other system administrators.

[Viewing and Managing Reports](#) (page 221)

2.5.1 Report Descriptions

PureMessage offers pre-defined reports that provide graphical representations of key performance statistics. To use these reporting features, the PostgreSQL database must be enabled. The reports provided by PureMessage are:

Attachment Sizes

Shows a histogram (a visual representation of the distribution) of the attachment sizes in a logarithmic distribution.

Attachment Types

Shows a count of attachments received, grouped by attachment type.

Note

By default, neither the Attachment Sizes nor the Attachment Types report contains data. If you want PureMessage to display reports data for attachments, edit the PureMessage policy, as described in “Adding Attachment Information” in the *Administrator's Reference*.

Message Categorizations

Shows the number of messages detected as spam, virus or other (neither virus nor spam). If PureMessage determines that a message contains spam and also contains a virus, the message counts toward the virus total only. By default, a message is marked as spam if its spam probability score is 50% or greater.

Rejected MTA Connections

Shows the number of connections rejected due to MTA-level IP blocking.

Overall Spam and Virus Count

Shows the total number of spam messages received and the total number of virus-infected messages received.

Messages from Blocked IPs in Policy

Shows the number of messages that matched the “Message is from Blocked IP” policy test. These messages appear in the report as “Blocked” even if a different policy action was used in the rule.

Policy Mark Hits

Shows a count of keys from the message log. If log-marking actions have been added to specific policy rules, this report can be used to count how many times those rules are triggered .

Quarantine Size

Shows the number of new messages added, helping you understand the growth rate of your quarantine. This report also shows the current size (in kilobytes) of local quarantines, and the total number of messages quarantined.

Number of Releases

Shows the number of messages released. This helps you to quickly understand current end user activity and look for any significant trends in the number of messages being released from the quarantine (which may indicate a need for filter tuning).

Rule Hit Rates

Shows the frequency of various spam rule hits. You may want to use this report if you adjust the weighting on a specific spam rule to tune the system for your environment. These rules are contained in the PureMessage updates you receive from Sophos.

Spam Range Volumes

Shows the number of messages by spam probability range, which is shown as a percentage. This report can help you tune your policy configuration to minimize the amount of spam getting quarantined or passed through to end users. It breaks down the scoring of potential spam into bands.

Top Other Relays

Shows the top spam relays by number of other messages (those that are classified as neither spam nor virus). This report can help you fine-tune your spam filtering performance by highlighting other relays (for example, partners) you may want to add to your whitelisted hosts.

Top Relays

Shows the top relays by number of messages.

Top Releasers:

Shows the top releasers of messages.

Top Spam Recipients

Shows the top spam recipients by number of detected spam messages. This report can help you understand which users receive large volumes of spam, allowing you to create custom policy rules to more aggressively filter spam for a group of users or a specific individual.

Top Spam Relays

Shows the top spam relays by number of detected spam messages.

Top Spam Senders

Shows the top spam senders by number of detected spam messages.

Top Virus Relays

Shows the top spam relays by number of detected virus messages.

Top Virus Types

Shows the virus types (categorized by virus ID) found in messages. For details on recent viruses by ID, see the Sophos "Latest virus identities" page.

Related concepts

[Adding Attachment Information](#) (page 303)

2.5.2 Viewing Reports

1. On the sidebar, click **Reports**.
The **Reports** page is displayed.
2. Click the name of the report that you want to view.
A report-specific page is displayed with the selected report in the content pane.

2.5.3 Modifying Reports

To modify the presentation of report data, or to modify the data ranges for an individual report, use the **Modify Report** options on the sidebar:

1. In the **Format** drop-down list, select a report format option.

2. In the **Time** drop-down list, select a time period option or select **Use Start/End Times** to set specific date and time parameters.
3. If you clicked on **Use Start/End Times** in the preceding step, set the year, month, day and time of the start and end times for the report.
4. In the **Grouping** drop-down list, click a server combination.
5. Click **Change**.

The selected graph with the specified parameters is displayed in the content pane

2.5.4 Exporting Reports

To export reports in comma separated value (CSV) format:

1. Open the individual page for the report whose data you want to export.
2. Set the desired reporting time and grouping.
3. On the sidebar under **Tasks**, click **Export**.

2.5.5 Scheduling Reports

1. Open the individual page for the report that you want to mail.
2. Set the desired reporting format, time and grouping.
3. On the sidebar under **Tasks**, click **Schedule**.

The **Schedule Reports** page is displayed.

4. In the **Main To** text box, enter the email address of the recipient.
5. In the Subject text box, enter the subject line for the email.
6. Below the Schedule form, click Schedule.

The **Add Job Entry** page is displayed.

Related information

[pmx-reports-mailer](#)

2.6 Local Services Tab

The **Local Services** tab provides an interface for managing the network services and the scheduled jobs running on the local system. These services and scheduled jobs make up PureMessage's main capabilities.

The **Background Services** table provides a list of the network services that are installed on the local system, as well as information about their status. The table contains controls to **Start**, **Restart**, and **Stop** these services. You can also click on the name of the service to view its status page. These per-service status pages show more detailed information on the selected services, and the sidebar on each page often contains additional links to configuration pages for that service.

The **Scheduled Jobs** table lists the regularly run commands that perform many of the jobs required to keep PureMessage running effectively. These jobs typically involve quarantine updates, quarantine digest generation and mailouts, and report data collection. The **Scheduled Jobs** table provides controls to add, modify enable, disable, or delete scheduled jobs.

The **Local Services** tab sidebar also provides access to a number of pages in where you can view global log files or set global configuration options.

2.6.1 Managing Background Services

Only the services and scheduled jobs that are included in the PureMessage roles that you have installed are displayed in the **Background Services** table. Also, the **AntiVirus Service** is only displayed in this table if it has been configured to run as a service.

The services that can be displayed in the **Background Services** table are described in the sections that follow.

Using the Background Services Table

- To manage services:
 - a) Select the check box to the left of the service that you want to manage.
 - b) Click the button at the bottom of the table for the service management operation that you want to perform, **Start**, **Restart**, or **Stop**.
- To view the status of a service:
 - a) In the **Service Type/Name** column, click the name of the service whose status you want to view.

Managing the Anti-Virus Service

By default, the Sophos Anti-Virus back end is loaded by the PureMessage engine (**Milter (Policy)** service) directly. Sophos Anti-Virus will only appear as a separate service if you select the **Run As Service** option on the **Policy: Anti-Virus Options** page in the PureMessage Manager. This adds an **AntiVirus Service** item on both the **Dashboard** tab and the **Local Services** tab.

The status, the process ID, the number of scanner processes, connections, and the queued messages are displayed in the **Details** column. You can stop or start the service from this page. For additional information, click **AntiVirus Service**.

You can access a service-specific configuration page by clicking **AntiVirus Service** in the **Background Services** table, and then clicking **Edit**. The **Anti-Virus Service** page displays the following information in the **Status** table:

- Status: Whether or not the service is running.
- PID: The Unix/Linux Process ID.
- Port: The port number used to communicate with the Anti-Virus service. By default, this is 18080.
- Connections: The current number of connections to this service.
- Queued: The number of messages queued for processing by the anti-virus service.
- Scanners: The number of anti-virus scanners that are running.

Note

You can also click **Edit** below the **Status** table to change the **Anti-Virus Service** options.

To manage the Anti-Virus Service: Click **Start**, **Restart**, or **Stop** to perform these actions. Only the actions that are currently available are displayed.

Changing the Anti-Virus Service Settings

1. Change the text boxes in the **Settings** table as required.

The text boxes are:

- **Concurrency Limit:** Specifies how many concurrent requests the anti-virus daemon can service at any given time. The default is the number of CPUs multiplied by 2.
- **Inactive Limit:** Specifies the maximum amount of time that a daemon slot is allowed to be inactive before it is removed and its memory is freed. The value must be lower than the time set in **Scan Limit** to be useful. The default setting is 2 minutes (120).
- **Scan Limit:** Specifies the absolute maximum amount of time to wait for the daemon to scan a message before returning a timeout. The default is 5 minutes (300).
- **Port:** Specifies the port used by PureMessage to communicate with the daemon. The default is `local:/tmp/pmx-vscan.sock`.
- **Reinit Limit:** Specifies the maximum number of messages each anti-virus process may scan before being restarted. Starting a process has a large overhead compared to the incremental cost of scanning an additional message. The value '0' means there is no limit, and the daemon only restarts when it times out or crashes. The default is 1024.
- **Queue Limit:** Specifies the maximum number of messages to queue before refusing to accept connections from PureMessage. Increasing this number too much means the daemon may consume large amounts of memory for its queue. The value '0' means there is no limit. The default is 4096.
- **Respawn Wait:** Specifies how long the daemon should wait before respawning a process that dies unexpectedly. If the process dies without scanning a message, the daemon waits for this length of time to elapse before restarting it. The default is 5 minutes (300).

2. Once you have made the required changes, click **Save**.

Managing the HTTPD (Manager) Service

The **HTTPD (Manager)** service provides the graphical user interface (GUI) for the PureMessage Manager. The **HTTPD (Manager)** service will only appear in the **Background Services** table if it has been installed. The status, the process ID, the port number (by default, 18080) are displayed. The **Details** column indicates whether SSL is enabled. For additional information, click **HTTPD (Manager)** to view the **Manager Status** page, which also shows the hostname on which this service is running, and the service's health.

The **Manager Status** page displays the following information in the **Manager Service** table:

- **Status:** Whether the service is running.
- **SSL:** Whether the secure socket layer protocol is enabled.
- **Port:** The port number used to communicate with this service. By default, this is 18080.
- **PID:** The Unix/Linux Process ID.
- **Hostname:** The DNS name of the server that is running the service.

To manage the HTTPD (Manager) Service: Click **Start** or **Restart** to perform these actions. Only currently available actions are displayed.

Related information

[pmx-manager](#)

Configuring SSL

If the Secure Sockets Layer (SSL) package is installed, SSL-encrypted communication between your web browser and the server where the PureMessage Manager is running is enabled by default. Using SSL is advised whenever confidential information (such as passwords) is transmitted over the internet. However, you must disable SSL if you do not have an SSL-compliant browser or if there is a firewall blocking HTTPS requests from your browser to the PureMessage Manager host.

To change the default SSL encryption settings:

1. Change the text boxes in the **SSL Support** table as required.

The text boxes are:

- Enable SSL if available: Use the drop-down list to enable or disable SSL support.
- Private Key File: Enter the path and filename of the private key .pem file.
- Certificate File: If the certificate file is the same as the private key file, select **Same file as private key**. Otherwise, enter the path specifying the location of the certificate file in the text box provided.

2. Once you have made the required changes, click **Save**.

Related tasks

[Generating a Self-Signed Certificate](#) (page 155)

Related information

[pmx-cert](#)

Generating a Self-Signed Certificate

To make an SSL connection more secure, it is recommended that you use a private certificate. By default, PureMessage uses the `/opt/pmx6/etc/manager/manager.pem` certificate. If you prefer to generate your own self-signed certificate, you can do so using the `pmx-cert` command.

The following instructions assume that OpenSSL is installed on the system; see <http://www.openssl.org/> for more information.

To generate a self-signed certificate:

1. At the command line, log in as the PureMessage user (by default "pmx6").
2. Change to the `/opt/pmx6/etc/manager/` directory beneath the root PureMessage installation directory.
3. Generate a `pmx-cert.pem` file with the following command:

```
pmx-cert --dns=<fully qualified domain name of the server>
--email ="<Administrator's email address>" --ip=<IP address of the
server>
--url=http://<fully qualified domain name of the server>
```

4. On the PureMessage Manager's **Local Services: SSL Encryption** page, edit the **Private Key File** text box so that it reads `/opt/pmx6/etc/manager/pmx-cert.pem`. Then ensure that the **Same file as private key** option button is selected, and click **Save**. You are prompted to accept the certificate. Select the option to accept it permanently.
5. On the **Local Services: Manager Status** page, click **Restart**.

Related tasks

[Configuring SSL](#) (page 154)

Related information

[OpenSSL Website](#)

Configuring Server IP Access Control

As added security, you can determine which systems are permitted to access the PureMessage Manager web server by specifying either hostnames or IP addresses for which access is allowed or denied. This is an important measure if the PureMessage Manager web server is accessible from the internet. Without this protection, anyone who guesses your password will have complete control of your PureMessage system.

1. On the **IP Access Control** page, enter hostnames (for example, `foo.bar.com`), IP addresses (for example, `255.255.255.128`) and IP networks (for example, `10.254.3.0` or `10.254.1.0/255.255.255.128`) in the text box.
2. Select the desired option button to **Allow From All Addresses** or **Only Allow From Listed Addresses** or **Deny From Listed Addresses**.
3. If you specified hostnames in the access permission list, enable the **Resolve hostnames on every request** option.
Requests to the PureMessage Manager server are received in the form of IP addresses. Enabling this option causes hostnames to be logged instead of IP addresses.
4. Once you have made the required changes, click **Save**.

Configuring Logging

You can configure the PureMessage Manager to record a log of user actions and module usage in common log format (CLF). Logs are written to the file `/opt/pmx6/var/log/manager/activity_log`. Once logging is configured, you can use the **Browse Activity Log** feature to analyze the activity of individual users and/or modules.

To configure logging:

1. Edit the text boxes on the **Configure Logging** page as required.

The text boxes are:

- **Enable Logging:** Use the drop-down list to enable or disable logging.
- **Resolve Host Names:** Select to perform a reverse DNS lookup to determine the hostname based on the IP address. If selected, hostnames are logged instead of IP addresses.
- **By Users:** Select the names of users for which logging is enabled. By default, the **All Users** check box is selected.
- **Clear Logfiles:** Select a time period from the drop-down list to specify the frequency with which the log file is cleared. Choose **Never** or **Every Week** or **Every Month**.
- **Log File Changes:** Select to have the log to include details about changed files and commands executed.
- **In Modules:** Select the names of modules for which logging is enabled. By default, the **All Modules** check box is selected.

Note

The modules listed match the PureMessage Manager tabs with two exceptions: **RPC** is the Remote Procedure Calls module, which allows commands to be run between remote systems and is used in some EUWI and CSM operations; **miniserv** is the web server used to serve the Manager web pages.

2. Once you have made the required changes, click **Save**.

Related information

[pmx-log](#)

Browsing the Activity Logs

If logging has been enabled, you can use the **Browse Activity Log** feature to search the PureMessage Manager log for details about the activity of the users and modules.

To review the activity of one or more users:

1. In the **By user** list box, select the user or users whose activities you want to review.
You can **Ctrl+Click** to select multiple user names, or you can select the **All users** check box.

2. In the **In module** list box, select the module or modules whose use by the selected user(s) you want to review.
You can **Ctrl+Click** to select multiple modules, or you can select the **All modules** check box.
3. Use the **From** and **To** drop-down lists to set the time period for the activity.
4. Once you have made the required changes, click **Search**.

The **Local Services: Activity Log** page is redisplayed with a table listing the results of your search.

5. Click the **Action** for an entry to view further details.

The **Activity Log** page is redisplayed with a table listing the details of the selected log entry. In the **Details of logged action** table, you can do the following:

- Click the **User** name to access the **Edit User** table for that user.
- Click the **Module** name to access the module in which the action was performed.
- Click the **URL** to access the page in which the action was performed.
- Click the **Session ID** to return to the **Search Results** table.

Related information

[pmx-log-summary](#)

[pmx-mlog-watch](#)

Creating Administrator Accounts

Access to the Manager is controlled by profiles for individual users and groups of users, which include access permissions to the various PureMessage modules.

To create a PureMessage Manager administrator account:

1. On the sidebar of the **Local Services** tab, click **New** beside **Administrators**.

The **Manager Status** page, containing the **Add User** table, is displayed.

2. In the **Add User** table, fill in the text boxes.

The text boxes:

- Name: Enter the full name of the user.
- Login: Enter the username for the account.
- Enabled: Select **Yes** to enable the account, or **No** to keep the account inactive.
- Password and Confirm Password: Enter and confirm the account's initial password.

Note

If, at some point, you choose to temporarily disable an administrator account, you must reset the password when you re-enable the account.

- Member of: Select the check boxes beside each group this administrator will be a member of. Click the name of the group to access the **Edit Group** page, which shows the access permissions that are granted to that group.
- Permissions: Select the desired check boxes to grant the user permissions in addition to the permissions granted by their group membership(s).

Note

The permissions mostly equate to which tabs (actually modules) that the user can access. The modules that differ from the available tabs are:

- View message content - Determines whether the user can view message content in the quarantine search results lists.
- RPC - Required to make any changes on remote systems on the **Server Groups** tab, such as starting, restarting, or stopping services on, and publishing configuration settings to, remote systems.
- RPC: view status - Lets users view information on remote systems On the **Server Groups** tab, but does not permit them to make changes.
- miniserv - This module must be checked to enable the next option, "Session authentication never expires".
- Session authentication never expires - This should always be granted, or the user's ability to log on will expire.

3. Once you have created the user profile, click **Save**.

The username of the created account is displayed in the **Administrators** section of the sidebar.

Related tasks

[Editing Administrator Accounts](#) (page 158)

[Editing Administrator Accounts](#)

1. In the **Administrators** section of the **Local Services** sidebar, click the name of the administrator account that you want to edit.

The **Manager Status** page, containing the **Edit User** table, is displayed.

2. Change the information for the selected account as required.

The text boxes that can be changed are:

- Name: You can change the full name of the user.
- Enabled: You can change whether the account is enabled or not.
- Password and Confirm Password: You can reset the admin user's password.

Note

If, at some point, you choose to temporarily disable an administrator account, you must reset the password when you re-enable the account.

- Member of: You can change which groups the user is a member of. Click the name of the group to access the **Edit Group** page, which shows the access permissions that are granted to that group.
- Permissions: You can change the user permissions that are granted in addition to the permissions granted by group membership(s).

Note

The permissions mostly equate to which tabs (actually modules) that the user can access. The modules that differ from the available tabs are:

- View message content - Determines whether the user can view message content in the quarantine search results lists.
- RPC - Required to make any changes on remote systems on the **Server Groups** tab, such as starting, restarting, or stopping services on, and publishing configuration settings to, remote systems.
- RPC: view status - Lets users view information on remote systems on the **Server Groups** tab, but does not permit them to make changes.
- miniserv - This module must be selected to enable the next option, "Session authentication never expires".
- Session authentication never expires - This should always be granted, or the users ability to log on will expire.

3. Once you have finished making changes, click **Save**.

Related tasks

[Creating Administrator Accounts](#) (page 157)

[Creating Administrator Groups](#)

1. On the sidebar of the **Local Services** tab, click **New** beside **Groups**.
The **Manager Status** page, containing the **Add Group** table, is displayed.
2. Fill in the text boxes in the **Add Group** table.

The text boxes are:

- Group: Enter a name for the group.
- Permissions: Select the check boxes beside the modules to which you want to grant group access.

Note

The permissions mostly equate to which tabs (actually modules) that the user can access. The modules that differ from the available tabs are:

- View message content - Determines whether the user can view message content in the quarantine search results lists.
- RPC - Required to make any changes on remote systems on the **Server Groups** tab, such as starting, restarting, or stopping services on, and publishing configuration settings to, remote systems.
- RPC: view status - Lets users view information on remote systems in the **Server Groups** tab, but does not permit them to make changes.
- miniserv - This module must be checked to enable the next option, "Session authentication never expires".
- Session authentication never expires - This should always be granted, or the users ability to log on will expire.

3. Once you have set the group name and permissions, click **Save**.

The name of the created group is displayed in the **Groups** section of the sidebar.

Related tasks

[Editing Administrator Group Accounts](#) (page 160)

[Editing Administrator Group Accounts](#)

1. In the **Administrators** section of the **Local Services** sidebar, click the name of the administrator account that you want to edit.

The **Manager Status** page, containing the **Edit User** table, is displayed.

2. Change the information for the selected account as required.

The text boxes that can be changed are:

- Group: You can change the name of the group.
- Permissions: You can change the group permissions.

Note

The permissions mostly equate to which tabs (actually modules) that the user can access. The modules that differ from the available tabs are:

- View message content - Determines whether the user can view message content in the quarantine search results lists.
- RPC - Required to make any changes on remote systems on the **Server Groups** tab, such as starting, restarting, or stopping services on, and publishing configuration settings to, remote systems
- RPC: view status - Lets users view information on remote systems on the **Server Groups** tab, but does not permit them to make changes.
- miniserv - This module must be selected to enable the next option, "Session authentication never expires".
- Session authentication never expires - This should always be granted, or the users ability to log on will expire.

3. Once you have finished making changes, click **Save**.

Related tasks

[Creating Administrator Groups](#) (page 159)

Managing the HTTPD (RPC/UI) Service

The **HTTPD (RPC/UI)** service provides the graphical user interface seen by end users of PureMessage and the Groups Web Interface. The **HTTPD (RPC/UI)** service only appears in the **Background Services** table if the option to enable both the End User Web Interface (*EUWI*) and the *Groups Web Interface* was selected during installation.

The EUWI provides individual user access to PureMessage mail-filtering features. End users can view and manage quarantined messages, manage user-specific whitelists and blacklists, and configure personal mail-filtering options.

The Groups Web interface allows a global administrator to delegate administrative responsibilities to "group" administrators based on groups/domains and/or roles. Delegated tasks can include quarantine management, reporting, list management and the configuration of certain policy settings.

The status, process ID, and the port number (by default, 28443) are shown in the **Details** column of the **Background Services** table.

For additional information, click **HTTPD (RPC/UI)** to view the **HTTPD (RPC/UI): Status** page, which also shows the hostname of the server that is running this service, the URL for the End User Web Interface, its health, and whether the service's configuration is current.

The **HTTPD (RPC/UI): Status** page displays the following information in the **RPC/UI Status** table:

- Status: Whether the service is running.
- Summary: The PureMessage package name that provides this service and the port used to access it.
- Servername: The fully-qualified domain name of the server running this service.
- Url: The full URL used to access this service.
- Ready: The readiness state of this service.

- **Config:** Whether or not the configuration of this service is current.
- **Port:** The port number used to communicate with this service. By default, this is 28080.
- **PID:** The Unix/Linux Process ID.

To manage the *HTTPD (RPC/UI) service*: Click **Start**, **Restart**, or **Stop** to perform these actions. Only currently available actions are displayed.

Configuring Access to the HTTPD (RPC/UI) Service

By default, this feature uses the **Host Access List**. To modify this list, or to create a new list for use in this feature, see "Editing Lists" in the Policy Tab section of the *Manager Reference*.

The RPC (Remote Procedure Call) service is the mechanism used by the End User Web Interface (EUWI) to communicate with the PureMessage Manager. Access to the HTTPD (RPC/UI) service is, by default, set to use the **Host Access List** as the source of the IP addresses or hostnames of authorized machines. This list should contain the IP addresses or hostnames of all of your PureMessage servers. You can use the **Local Services: Configure RPC Service** page to choose which list is used to determine access control.

To set the RPC service's host access list:

1. In the **Background Services** table of the **Local Services** tab, click **HTTPD (RPC/UI)**, and then, on the sidebar, click **Configure RPC Service**.

The **Configure RPC Service** page is displayed.

2. In the **RPC** table, select the list that you want to use as the **Host Access List** from the drop-down list.
3. Click **Save**.

Related tasks

[Editing Lists](#) (page 120)

Related information

[pmx-rpc-enduser](#)

Configuring the RPC Server Location

To configure the location and access timeout of the RPC server:

1. In the **Background Services** table on the **Local Services** tab, click **HTTPD (RPC/UI)**, and then click **Configure End User UI** on the sidebar.

The **Configure End User UI** page is displayed.

2. Change the text boxes in the **Configuration** table as required.

The text boxes are:

- **RPC Server:** Specifies the address and port number of the server that is running the End User Web Interface.
 - **RPC Timeout:** Specifies the amount of time that must elapse before the connection to the HTTPD (RPC/UI) server times out. The default is one minute (1m). Use the suffix "m" or "s" to indicate minutes or seconds, respectively. If no time suffix is specified, seconds (s) is assumed.
3. Once you have finished making changes, click **Save**.

Related information

[pmx-rpc-enduser](#)

Setting the End User Web Interface Server

To set the hostname and port of the server that is running the End User Web Interface:

1. In the **Background Services** table on the **Local Services** tab, click **HTTPD (RPC/UI)**, and then on the sidebar, click **Configure HTTPD Service**.

The **Configure HTTPD Service** page is displayed.

2. Change the text boxes in the **Configuration** table as required.

The text boxes are:

- **Hostname:** The hostname must be a fully qualified domain name.
- **Port:** The default is 28080. This can be changed to any port number greater than 1024.

3. Once you have finished making changes, click **Save**.

Related information

[pmx-httpd](#)

Generating a Self-Signed Certificate for the End User Web Interface

To make an SSL connection more secure, it is recommended that you generate your own self-signed certificate. The following instructions assume that OpenSSL is installed on the system; see <http://www.openssl.org/> for more information.

To generate a self-signed certificate:

1. At the command line, log in as the PureMessage user (by default "pmx6").
2. Change to the `/opt/pmx6/etc/manager/httpd2/` directory beneath the root PureMessage installation directory.
3. Back up `pmx-cert.cert` and `pmx-cert.pem` by running the following commands:

```
mv pmx-cert.cert backup-pmx-cert.cert
mv pmx-cert.pem backup-pmx-cert.pem
```

4. Generate a new `pmx-cert.pem` file with the following command:

```
pmx-cert --dns=<fully qualified domain name of the EUWI server>
--email ="<Administrator's email address>" --ip=<IP address of the
EUWI server>
--url=http://<fully qualified domain name of the EUWI server>
```

5. Ensure that the `SSLCertificateFile` option in the `/opt/pmx6/etc/manager/httpd2/ssl.conf` file is set to `/opt/pmx6/etc/manager/httpd2/pmx-cert.pem`
6. Restart the HTTP (RPC/UI) service with:

```
pmx-httpd stop; httpd
```

Related tasks

[Configuring SSL](#) (page 154)

Related information

[OpenSSL Website](#)

[pmx-cert](#)

Managing the Log Search Index Service

This service indexes new log messages as PureMessage processes mail, allowing for quicker log searches when using the log search options in the [Groups Web Interface](#). The service consumes messages from both the mail and message logs. This service must be enabled before you can use the log search options in the Groups Web Interface. Because the Postfix mail transfer agent is required

for log searches, this service is disabled by default for installations using sendmail. It is also disabled by default for Postfix users who have upgraded to the latest version of PureMessage from a previous version of PureMessage.

Important

Before enabling the Log Search Service, you must rotate both the message log and the mail log. See “Rotating PureMessage Log Files” for more information.

To enable the Log Search Service: If you have upgraded to the latest version of PureMessage from version 5.x, you must manually enable the service in `/opt/pmx6/etc/logsearch.conf` by setting `enabled` to `yes`. Then, on the **Log Search Index: Status** page, click **Start**.

The **Log Search Index: Status** page shows the following information in the **Log Search Index Status** table:

- **Status:** Whether the service is running.
- **PID:** The Unix/Linux Process ID.
- **Ready:** Whether the service is ready for indexing.

To manage the Log Search Index Service: Click **Start**, **Restart**, or **Stop** to perform these actions. Only currently available actions are displayed.

Related concepts

[Log Search](#) (page 218)

[Rotating PureMessage Log Files](#) (page 38)

Managing the IP Blocker Service

The **IP Blocker** service rejects messages originating from IP addresses blacklisted by Sophos Labs. Enabling this option can improve performance by blocking spam before it reaches more complex tests in the policy.

Note

Whether you choose to block IP addresses by enabling MTA-level IP blocking or by using the PureMessage policy, PureMessage requires that the IP Blocker Service be enabled.

The **IP Blocker Service: Status** page shows the following information in the **Blocker Status** table:

- **Status:** Whether the service is running.
- **Summary:** The PureMessage package name that provides this service and the port used to access it.
- **Config:** Whether the configuration of this service is current.
- **Port:** The port number used to communicate with this service.
- **PID:** The Unix/Linux Process ID.

To manage the IP Blocker Service: Click **Start**, **Restart**, or **Stop** to perform these actions. Only currently available actions are displayed.

Managing the Milter (Policy) Service

The **Milter (Policy)** service is the core mail-processing service for PureMessage. It accepts messages from a mail transfer agent (MTA), processes them according to configured policies, and then passes the messages back to the MTA for delivery.

The **Milter (Policy)** service is always displayed in the **Background Services** table, along with its status and process ID. The number of milter processes are shown in the **Details** column. For additional information, click **Milter (Policy)** to view the **Milter Status** page, which also shows the number of threads, the port on which this service is running, and the Virtual Memory (VM) size of the top-level process.

The **Milter Status** page shows the following information in the **Milter: Policy** table:

- **Status:** Whether the service is running.
- **Port:** (with sendmail) The port number at which the Milter communicates with the Sendmail service.
- **SMTP Listen Port:** (with Postfix) The port number at which the Milter service listens for connection requests.
- **SMTP Talk Port:** (with Postfix) The port number used by the Milter service to communicate with Postfix.
- **Process Id:** The Unix/Linux Process ID.
- **VM Size:** The virtual memory size being used.
- **Threads:** The number of available threads, if running in threaded mode (this may be blank).
- **Processes:** The number of milter service processes that are running.

To manage the Milter service: Click **Start**, **Restart**, or **Stop** to perform these actions. Only currently available actions are displayed.

Changing Milter Configuration Options

1. In the **Background Services** table on the **Local Services** tab, click **Milter (Policy)**, and then click **Milter Options** on the sidebar.

The **Milter Options** page is displayed.

2. Change the text boxes in the **Interpreters** table as required.

The text boxes are:

- **Concurrent Interpreter Limit:** Specifies how many concurrent requests PureMessage can service at any given time. The default is calculated during installation, based on the physical memory available on the system.
- **Concurrency Limit Action:** Specifies the action to take if the concurrency limit is reached. Valid option values are 'wait', 'tempfail' or 'accept'. The default is 'tempfail'.
- **Preload Interpreters:** Specifies how many perl interpreters should be started for a multithreaded milter. The default is the number of CPUs on the system multiplied by five.
- **Interpreter Reinitialization (number of connections):** Specifies how many connections a perl interpreter will serve before it is reinitialized. A value of 0 means never reinitialize. The default is 2048.
- **Use Threads:** Disabling this option makes the milter(s) fork new processes to handle new connections instead of using a new thread. The default is "no" on all platforms.
- **Milter umask:** Sets the umask value for the milter process. The default is 007.
- **Debug Output Level:** Selects the verbosity in the milter log. Higher levels result in more verbose logging. The default is "No Debug Output", which means that log messages with DEBUG priority are suppressed.

- **Message Buffer Size:** Sets the internal buffer size that can be used to hold each message. Messages bigger than this are temporarily written to disk. The default is 1,000,000 (roughly 1 MB).
- **Maximum Requests Allowed Before Removal:** When the process pool is enabled, this option specifies how many connections a process is allowed to handle before it stops and a new process can take its place. The default is 0, which means that processes are not retired.
- **Maximum Time Idle Before Removal From Pool:** When the process pool is enabled, this option specifies how long a process in the pool can stay idle before it goes away. The default is 5m.
- **Log Connection Time:** If enabled, a log entry is written to the milter log for the time spent on each connection. The default is "off".
- **Log Message Processing Time:** If enabled, a log entry is written to the milter log for the time spent on each message. The default is "off".
- **Timestamp Logs in GMT:** Specify whether the log timestamps use GMT time or local time. The default is "no".

3. Click **Save**.

Related information

[pmx.conf](#)

[pmx-milter](#)

Viewing Milter Activity Logs

1. In the **Background Services** table on the **Local Services** tab, click **Milter (Policy)**, and then click **Activity Log** on the sidebar.

The **Milter: Activity Log** page is displayed.

2. Set the log file viewing filters in the **Recent Activity** table as required.

The text boxes are:

- **Log:** Select the log you wish to view. There are three logs available:
 - **Activity Log:** Display messages processed by the PureMessage milter.
 - **Error Log:** Display errors reported by the milter.
 - **Output Log:** Displays milter output.
- **Priority:** Select the priority level of the log entries that you wish to view. The log priority levels correspond to those used in syslog.
- **Auto Refresh:** If this option is enabled, the display is updated every 5 seconds.
- **Filter:** Enter full or partial data from any of the displayed text boxes to filter the output based on that data.
- **Show:** Enter the number of log records to display. The most recent log records are displayed.
- **Collapse:** If this option is enabled, multiple entries with the same Queue ID are condensed into a single entry.

3. Once you have set your preferred viewing options, click **View** to see the results.

Managing the PostgreSQL Service

The PostgreSQL service is the database from which PureMessage reports are generated. PostgreSQL also stores user resource data and often serves as the quarantine database. The PostgreSQL service will only appear in the **Background Services** table if PostgreSQL is installed. The service status and process ID are displayed in the **Background Services** table. The port that the service uses is shown in the **Details** column.

For additional information, click **PostgreSQL Service** to view the **PostgreSQL Service: Status** page, which also shows the service's state of readiness, and the host on which the PostgreSQL service is running.

The **PostgreSQL Service: Status** page displays the following information in the **PostgreSQL Status** table:

- Status: Whether the service is running.
- Summary: A summary of the major PureMessage features using the PostgreSQL database service and the port number they use to access it.
- Ready: The readiness state of the service.
- Port: The port number used to communicate with the service.
- Hostname: The hostname on which the service is running.
- PID: The Unix/Linux Process ID.

*To manage the **PostgreSQL Service**:* Click **Start**, **Restart**, or **Stop** to perform these actions. Only currently available actions are displayed.

Managing the Queue Runner Service

The Queue Runner service manages and continuously flushes the queue, a folder in the message store that holds messages that are about to be injected back into the mail system. The **Queue Runner** service is always displayed in the **Background Services** table.

The **Queue Runner** service page displays the following information in the **Status** table:

- Status: Whether the service is running.
- PID: The Unix/Linux Process ID.
- Ready: The readiness state of this service.

*To manage the **Queue Runner** service:* Click **Start**, **Restart**, or **Stop** to perform these actions. Only currently available actions are displayed.

Managing the Scheduler Service

The Scheduler service is a cron replacement that gives you the added control of being able to easily stop and graphically manage all scheduled PureMessage jobs. The **Scheduler Service** is always displayed in the **Background Services** table. The jobs that the Scheduler service runs are shown in **Scheduled Jobs** table below.

The **Scheduler Service** page displays the following information in the **Status** table:

- Status: Whether the service is running.
- PID: The Unix/Linux Process ID.

*To manage the **Scheduler Service**:* Click **Start**, **Restart**, or **Stop** to perform these actions. Only currently available actions are displayed.

Related tasks

[Managing Scheduled Jobs](#) (page 172)

Managing the SMTP (Postfix) Service

The SMTP (Postfix) service is one of the mail transfer agents that PureMessage supports. The **SMTP (Postfix)** service will only appear in the **Background Services** table if the version of Postfix distributed with PureMessage was enabled during installation.

The Background Services table shows the status of the Postfix service and its process ID. For additional information, click **SMTP (Postfix)** to view the **Postfix Status** page, which shows Postfix's state of readiness and the Virtual Memory (VM) size of the Postfix process.

The **SMTP (Postfix)** service page displays the following information in the **Status** table:

- Status: Whether the service is running.
- PID: The Unix/Linux Process ID.
- Ready: The readiness state of the service.
- VM Size: The current virtual memory size.

To manage the SMTP (Postfix) service: Click **Start**, **Restart**, or **Stop** to perform these actions. Only currently available actions are displayed.

Configuring Postfix Content Filtering

PureMessage will work with any version of Postfix that has content-filtering enabled. The Postfix content-filtering mechanism relies on the transfer of messages via SMTP to PureMessage. PureMessage can then modify and/or reinject the message back into Postfix, quarantine the message, or take other actions based on the PureMessage policy.

To modify the Postfix content-filtering configuration:

1. In the **Background Services** table on the **Local Services** tab, click **SMTP (Postfix)**.

The **Postfix Status** page is displayed.

2. On the sidebar, click **Content Filtering Configuration**.

The **Postfix Content Filtering** page is displayed.

3. Change the text boxes in the **Settings** table as required.

The text boxes are:

- Host: The interface on which PureMessage listens for Postfix connections. If PureMessage is running on the same host as Postfix, this should be set to localhost. If not, specify the fully qualified hostname or IP address of the host where PureMessage is running. The value used here should correspond to the host part of the SMTP Talk Port (see "Editing the Basic Milter Settings" for more information).
- Port: The port on which PureMessage listens for incoming Postfix connections. The default is 10025. The value used here should correspond to the port part of SMTP Talk Port (see "Editing the Basic Milter Settings" for more information).
- Forward Host: The host through which PureMessage sends filtered mail. As with the Host option, this can be set to localhost if PureMessage and Postfix are running on the same host. The value used here should correspond to the host part of SMTP Listen Port (see "Editing the Basic Milter Settings" for more information).
- Forward Port: The port on which PureMessage sends the message back to Postfix. The default is 10026. The value used here should correspond to the host part of SMTP Listen Port (see "Editing the Basic Milter Settings" for more information).
- Concurrency: Determines how efficiently the system schedules filtering tasks. Higher concurrency values generally imply less latency in processing messages, but incur more scheduling overhead and higher memory consumption. This value should be increased in proportion to the number of CPUs on the system. A value of 3 x CPUs is recommended, with a minimum of 10.

4. Click **Save**.

5. Re-start the Postfix service for the changes to take effect.

Reconfiguring the Postfix Service

1. In the **Background Services** table on the **Local Services** tab, click **SMTP (Postfix)**.

The **Postfix Status** page is displayed.

2. On the sidebar, click **Change Postfix Configuration**.

The **Postfix Configuration** page is displayed.

3. Change the text boxes in the **Settings** table as required.

The text boxes are:

- **Name of Mail Server:** Specify the fully qualified internet hostname of the mail server. For example, `myhost.example.org`.
- **Domain:** Specify the local internet domain name to which this server belongs. For example, `example.org`.
- **Originating Address:** Specify the domain name that Postfix should append to unqualified addresses. The value may be specified as `$myhostname` or `$mydomain` to use the values specified in the **Name of Mail Server** and **Domain** text boxes.
- **Destination Address:** Specify a list of email addresses that Postfix should deliver messages for locally. Values in the list should be separated by a comma and followed by a single space. Again, `$myhostname` or `$mydomain` can be used as in the **Originating Address** text box. For example:
 - For a workstation or gateway, use: `$myhostname, localhost.$mydomain`
 - For a mailserver for an entire domain (for example, a mail hub), use `$myhostname, localhost.$mydomain, $mydomain`
 - For a mail server that uses a virtual host interface, use `$myhostname`
- **Mail Relay:** If the machine is on an open network, then you must specify what client IP addresses are authorized to relay their mail through that machine. The default setting includes all class A, B or C networks that the machine is attached to. Often, that gives relay permission to too many clients, so specifying a more restrictive list is recommended.
- **Gateway:** The SMTP relay to use for mail delivery. If the machine is behind a firewall, or if the network has a dedicated mail gateway, specify the host to relay through. For example:
 - To use the server(s) in your domain's [DNS MX records](#), enter `$mydomain`
 - To use the server(s) associated with a host's MX records, enter `mail.$mydomain`
 - To use a specific host without paying attention to MX records, enter `[mail.$mydomain]`

4. Click **Save**.

5. Re-start the Postfix service for the changes to take effect.

Configuring Postfix for TLS Encryption

To configure [TLS](#) you will need the following

1. A self-signed or purchased CA certificate (`MyCert.pem`)
2. Certificate key (`MyCertKey.pem`)
3. The ROOT certificates from CA's you wish to trust (`CaCert.pem`)

To enable TLS within Postfix, as the root user:

1. Create a new directory named `/opt/pmx6/postfix/etc/certs/`
2. Place your certificates within `/opt/pmx6/postfix/etc/certs/` ensuring they are owned by the root user
3. Edit the file `/opt/pmx6/postfix/etc/main.cf`

4. Add the following to the end of the file:

```
# ----- Enable TLS -----
smtpd_use_tls = yes
smtpd_tls_key_file = /opt/pmx6/postfix/etc/certs/MyCertKey.pem
smtpd_tls_cert_file = /opt/pmx6/postfix/etc/certs/MyCert.pem
smtpd_tls_CAfile = /opt/pmx6/postfix/etc/certs/CaCert.pem
smtpd_tls_loglevel = 3
smtpd_tls_received_header = yes
smtpd_tls_session_cache_timeout = 3600s
tls_random_source = dev:/dev/urandom
# ----- END TLS -----
```

5. Save your `main.cf` and restart Postfix

```
# /opt/pmx6/postfix/etc/init.d/postfix restart
```

6. Check to see if TLS is enabled within Postfix

```
# telnet localhost 25
ehlo localhost
```

Postfix will advertise its capabilities similar to the following output:

```
C: [root@example.com]# telnet mail.example.com 25
S: 220 mail.example.com ESMTP Postfix (1.1.5)
C: EHLO example.com
S: 250-mail.example.com
S: 250-PIPELINING
S: 250-SIZE 10240000
S: 250-VRIFY
S: 250-ETRN
S: 250-STARTTLS
S: 250 8BITMIME
C: STARTTLS
S: 220 Ready to start TLS
```

Postfix now advertises TLS and can start a session.

Related information

Managing the SMTP (Sendmail) Service

The SMTP (Sendmail) service is one of the mail transfer agents (MTA) that PureMessage supports. The **SMTP (Sendmail)** service will only appear in the **Background Services** table if the version of sendmail distributed with PureMessage was enabled during installation.

The **Background Services** table shows the status and the process ID. It also indicates whether the service is configured (in the **Details** column). For additional information, click **SMTP (Sendmail)** to

view the **Sendmail Status** page, which also shows sendmail's state of readiness and the Virtual Memory (VM) size of the sendmail process.

The **Local Services: Sendmail Status** page displays the following information in the **Sendmail** table:

- **Status:** Whether the service is running.
- **Process Id:** The Unix/Linux Process ID.
- **Ready:** The readiness state of the service.
- **VM Size:** The virtual memory size being used.

To manage the SMTP (Sendmail) service: Click **Start**, **Restart**, or **Stop** to perform these actions. Only currently available actions are displayed.

Reconfiguring the Sendmail Service

1. In the **Background Services** table on the **Local Services** tab, click **View Milter Configuration**.

The **Milter Settings** page is displayed.

2. To change the settings, click **Edit** below the **Input Mail Filter Configuration: Policy** table.

The **Sendmail: Edit Milter** page is displayed.

3. Change the settings as required:

- **Status:** Specify whether the sendmail service is **Enabled** or **Disabled**.
- **Port:** By default, the milter socket address is `inet:3366@localhost`. This value must match the port on which the PureMessage milter is configured to run. Once you have set the port, click **Ping** to confirm that the port is properly set.
- **Fail mode:** Specify the action that sendmail will perform if it fails to connect to the PureMessage milter. The options are:
 - **Accept:** Sendmail will deliver the message anyway.
 - **Tempfail:** Sendmail will pause temporarily, then retry later. (default option)
 - **Reject:** Sendmail will reject the message.
- **Timeouts:** Timeout settings determine the amount of time the sendmail milter waits for various actions before it fails. Milter timeouts are described in detail in the topic "What timeout values do you recommend to use for filters?" in the PureMessage FAQ. Additional milter configuration information is included in the topic "Configuring milter parameters for sendmail" in the Sophos Knowledgebase.

4. Once you have made the required changes, click **Save**.

5. Restart the sendmail service to make the changes take effect.

Related information

[Configuring Milter Parameters for sendmail](#)

Running the Sendmail Configuration Wizard

To change how sendmail is configured to run (as a hub or a gateway), you must run the **Configuration Wizard**.

CAUTION

Changing the sendmail configuration to another mode erases any custom settings made to the `sendmail.mc` file. Any settings that configure or reference sendmail milters must be reapplied.

To run the Sendmail Configuration Wizard:

1. In the **Background Services** table on the **Local Services** tab, click **Run Configuration Wizard**.

The **Select Mail Server Type** page is displayed.

2. Select one of the two mail server types:

- Mail Hub: Select to deliver mail to all local accounts.

a) Click **Next**.

The **Configure a Mail Hub** page is displayed.

- b) Select the check box at the bottom of the **Save Configuration** table if you want the sendmail service automatically restarted when you finish.

c) Click **Finish**.

- Mail Gateway: Select to forward email to another server for redistribution.

a) Click **Next**.

The **Configure a Gateway Mail Server** page is displayed.

- b) Enter the information for each domain and server in the following manner.

Enter the domain name (for example, `myorganization.com`). To include all sub-domains, enter a period at the beginning of the domain name. (for example, `.myorganization.com`). Enter a space or a tab. Enter the hostname. Be sure to enclose the hostname in square brackets if you don't want to perform an MX lookup to resolve the host (for example, `smtp: [exchange.myorganization.com]`).

- c) Once you have specified all of the required servers and domain names, click **Next**.

The **Specify the domains that this server will relay email for** table is displayed.

- d) Specify addresses, subnets, hostnames, domain names or sub-domains.

Each entry must be on a separate line. Be sure to add `localhost` to the list.

e) Click **Next**.

The **Save Configuration** dialog box is displayed.

- f) Select the check box at the bottom of the **Save Configuration** table if you want the sendmail service automatically restarted when you finish.

g) Click **Finish**.

The **Sendmail Status** page is displayed.

3. If you did not select the option to restart sendmail automatically, you must restart the service manually for your changes to take effect by clicking **Restart** at the bottom of the **Sendmail** table.

2.6.2 Managing Scheduled Jobs

"Scheduled jobs" are cron replacements, commands scheduled to run at specified intervals. Various jobs are included as part of a PureMessage installation. Many are enabled by default. For detailed information about individual jobs, see the appropriate section of the *Administrator's Reference*, which contains descriptions of PureMessage commands, including those run as scheduled jobs. For a brief description of each default scheduled job, see "Scheduled Job Descriptions".

CAUTION

The scheduled jobs that enable the PureMessage Monitor are installed by default but are disabled. The PureMessage Monitor is an advanced tool that should only be implemented with the assistance of Sophos Support or Sophos Professional Services. Attempting to configure and implement this feature on your own may cause PureMessage to fail.

The **Scheduled Jobs** table displays the default scheduled jobs that are available in a new PureMessage installation and that have been configured to run automatically at set intervals by the

Scheduler Service. These jobs typically involve quarantine updates, quarantine digest generation and mailouts, and report data collection. You can add and schedule other jobs as well. The **Scheduled Jobs** table provides controls to add, modify enable, disable, or delete scheduled jobs.

- *To add a new scheduled job:* At the bottom of the **Scheduled Jobs** table, click **Add New** to open the **Add Job Entry** page.
- *To set or modify the schedule for a particular job:* Click the <Job Name> to open the **Edit Job Entry** page.
- *To enable, disable or permanently delete a scheduled service:* Select the check box beside the name of the service(s) you want to manage, and then click **Enable**, **Disable**, or **Delete** at the bottom of the **Scheduled Jobs** table.

Related concepts

[Scheduled Job Descriptions](#) (page 173)

Scheduled Job Descriptions

The default scheduled jobs are displayed in the lower portion of the status page of the **Local Services** tab. Not all jobs are enabled by default. The following jobs are displayed:

- **blocklist-compile:** Generates a blocklist file that combines data from Sophos Labs and customer lists. (CSM, EDGE, both enabled by default)
- **pmx-makemap :** Compiles a standard list or map and/or the group-specific list named members-per-group into a CDB list or map. (CSM, EDGE, both enabled by default)
- **pmx-mark :** Writes timestamp marks in log files. (EDGE, enabled by default)
- **pmx-mlog-stats:** Gathers data about the message log to provide Sophos with statistical feedback. It is recommended that you select the [Share data with Sophos](#) feature on the **Support** tab of the PureMessage Manager. Doing so will automatically enable the **pmx-mlog-stats** job. Providing these statistics to Sophos Labs gains you improved protection against spam threats. (CSM, EDGE, both disabled by default)
- **pmx-mlog-watch :** Scans the activity in the message log's in-bound traffic for irregularities that may indicate spam activity. (CSM, EDGE, both disabled by default)
- **pmx-prd-reduce-daily:** Reduces the last day's data in the PureMessage reporting database. (CSM enabled by default, EDGE disabled by default)
- **pmx-prd-reduce-hourly:** Reduces the last hour's data in the PureMessage reporting database. (CSM enabled by default, EDGE disabled by default)
- **pmx-prd-reduce-weekly:** Reduces the last week's data in the PureMessage reporting database. (CSM enabled by default, EDGE disabled by default)
- **pmx-prd-reduce-v2:** Reduces the data in the PureMessage reporting database's new reports tables. (CSM enabled by default, EDGE disabled by default)
- **resource-sync :** Synchronizes resources between local and database stores. (CSM, EDGE, both enabled by default)
- **pmx-qdigest :** Generates digests of quarantined messages. (CSM, EDGE, both disabled by default)
- **pmx-qdigest-expire :** Deletes old quarantine digests (those that exceed a specified age) stored in the `var/digest/pending` directory. (CSM, EDGE, both disabled by default)
- **pmx-qexpire :** A wrapper for **pmx-store-expire**, which can archive and delete messages from the message store that are older than the specified age. (CSM, EDGE, both enabled by default)
- **pmx-qmeta-index :** Add message metadata to the quarantine index. (CSM, EDGE, both enabled by default)
- **pmx-qrelease :** Process centrally stored quarantine action requests. (CSM, EDGE, both enabled by default)

- [pmx-quarantine-tidy](#) : A wrapper for `pmx-store`, which manages message stores, the indexed collections of messages organized into a directory structure, such as the messages in the PureMessage quarantine. (CSM, EDGE, both enabled by default)
- [consume-blocklist-log](#) : Consumes the data from the `blocklist_log` and uses it to populate the reports database. (CSM, EDGE, both disabled by default)
- [consume-message-log](#) : Consumes the data from the `message_log` and uses it to populate the reports database. (CSM, EDGE, both enabled by default)
- [consume-pmx-log](#) : Consumes the data from the `pmx_log` and uses it to populate the reports database. (CSM, EDGE, both enabled by default)
- [consume-quarantine](#) : Consumes the data from the quarantine and uses it to populate the reports database. (EDGE, enabled by default)
- [reports-time-ranges](#) : Populates the table `prd_period` with time ranges for reporting. (CSM, EDGE, both enabled by default)
- [pmx-webui-cleanup](#) : A utility that cleans up End Web User Interface sessions and temporary files. (CSM, enabled by default)

Important

By default, the following data update jobs are scheduled to run every five minutes. It is not recommended that you adjust these settings because doing so will compromise the level of best protection that is ensured by frequent Sophos Labs updates.

- `pmx-blocklist-data-update`: The Perl Package Manager (PPM), run to upgrade and install PureMessage blocklist data. (CSM, EDGE, both disabled by default)
- `pmx-antispam-data-update`: The Perl Package Manager (PPM), run to upgrade PureMessage anti-spam data. (CSM, EDGE, both enabled by default)
- `pmx-sophos-data-update`: The Perl Package Manager (PPM), run to upgrade PureMessage Sophos (anti-virus) data. (CSM, EDGE, both enabled by default)
- `pmx-db-watchdog`: Checks if PGSQL is alive. (CSM, enabled by default)

Scheduling a Job

1. In the **Scheduled Jobs** table, click **Add New**.

The **Add Job Entry** page is displayed.

Note

If you want to reschedule or edit an existing job, click the job's title in the **Scheduled Jobs** table instead of **Add New**.

2. On the **Add Job Entry** page, add the job command in the Command field if one does not exist. Or, optionally, modify the existing command.
3. In the Description text box, add a meaningful name for the job if one does not exist. Or, optionally, modify the existing description.
4. In the Schedule section, set the times at which you want the report emailed.
5. Select Enabled to activate the scheduled job.
6. Click Save.

You are returned to the page from which you opened the Status page, and the added or modified job is set to run on the selected schedule.

Scheduling Jobs in the Manager

When using the Scheduled Jobs interface in the PureMessage Manager to create the scheduled jobs, create jobs with the characteristics show in the following example:

Digest Generation

- Command: `/opt/pmx6/bin/pmx-qdigest`
- Description: Generate quarantine digests once a day at 1 PM
- Enabled: checked
- Schedule:
 - Hour: 13
 - Minutes: 00
 - Month: Any
 - Day: Any
 - Week Day: Any

Digest Expiry

- Command: `/opt/pmx6/bin/pmx-qdigest-expire`
- Description: Expire pending digests once a day at 11 PM
- Enabled: checked
- Schedule:
 - Hour: 23
 - Minutes: 00
 - Month: Any
 - Day: Any
 - Week Day: Any

Related concepts

[Scheduling Digest Tasks](#) (page 293)

Related tasks

[Managing Scheduled Jobs](#) (page 172)

Related information

[pmx-qindex](#)

[pmx-qdigest](#)

[pmx-qdigest-approve](#)

[pmx-qdigest-expire](#)

[pmx-qdigest.conf](#)

[pmx-qexpire](#)

2.6.3 Using the Local Services Tab Sidebar

The sidebar on the **Local Services** tab contains links to several pages in which you can view global log files or set global configuration options. These features are:

- **View Central Server Log:** Allows you to view the resource synchronization logs.
- **Edit Global Options:** Allows you to edit general service options.
- **Edit Log Watch Options:** Allows you to set options for the `pmx-mlog-watch` utility.
- **MTA IP Blocking:** Allows you to configure MTA level IP Blocking options.

- **Auto Refresh:** Allows you to enable and set the interval for automatically refreshing the status of background services and scheduled jobs.

Note

Clicking **View Services Status** in the **Tasks** section of the **Local Services** sidebar returns you to, or refreshes, the **Local Services: Status** page.

Viewing Central Server Logs

The **Local Services: Central Server Log Notes** page displays errors and events related to end user resources. End user resources are collections of configuration data that PureMessage components can utilize without concern for where that data is actually stored, such as in the database or in a configuration file. This allows configuration data to be shared between PureMessage servers.

The Central Server Log reports are organized by time and priority. The more severe priorities (critical, alert, and emergency) provide administrators with the first indication that there are problems with resources synchronizing across hosts. The less severe priorities provide information on successful resource synchronization operations.

You can perform the following tasks using the **Central Server Log Notes** page:

- *To limit the displayed log reports to only one priority:* Select a priority from the **Priority** drop-down list at the top of the reports table, and then click **Filter**.

Priorities are based on the UNIX/Linux syslog priority scheme, which is:

- EMERG - system is unusable
 - ALERT - action must be taken immediately
 - CRIT - critical conditions
 - ERR - error conditions
 - WARNING - warning conditions
 - NOTICE - normal but significant condition
 - INFO - informational
 - DEBUG - debug-level messages
- *To delete log reports:* Select the check box to the left of each report that you want to delete, and click **Delete** at the bottom of the reports table.
 - *To delete all log reports:* Click **Delete All** at the bottom of the reports table.
 - *To view other pages of reports:* Click the number of the page that you want to view to the right of **Go to Page** at the bottom of the reports table.

Related information

[pmx-csl](#)

Setting Global Options

The **Local Services: Global Options** page allows you to change the administrator mail settings, SMTP host and port settings, and the quarantine secret. These settings are saved in `/opt/pmx6/etc/pmx.conf`.

To set these global options:

1. Change the **Administrative Mail Settings** and **Quarantine Options** text boxes as required.

The text boxes are:

- **Administrator Name:** Specifies the PureMessage administrator's name, which is used in the From header of messages generated by PureMessage. For example, `PureMessage Admin`.
- **Administrator Email:** Specifies the PureMessage administrator's email address, which is appended to the Administrator Name in the From header of messages generated by PureMessage. For example, `PureMessage Admin [PureMessageAdmin@foo.com]`.
- **Mail Server (SMTP) Host:** Specifies the server to which re-sent messages are directed. For example, when messages are released from quarantine, they are routed via this server. The value is normally specified in the form `smtp:<hostname>` (for example `smtp:mail.foo.com`).
- **Mail Server (SMTP) Port:** Specifies the port on the server specified above (normally "25").
- **Quarantine Secret:** When messages are released from quarantine, an `X-PMX-Quarantine-Approved` header with the "Quarantine Secret" value is added to the message. When the PureMessage milter encounters this header, it validates the contents, and if it matches, the header is deleted and the message is delivered without filtering. There is no default; it is uniquely configured for each site during the PureMessage installation.

Note

It is important that you do not disclose this value because email that is formatted with the proper header and this key is not processed by PureMessage.

2. Once you have finished making changes, click **Save**.

Related information

[pmx.conf](#)

Setting Log Watch Options

Denial of service (DoS) and directory harvesting attacks are brute-force attacks that cause a sharp spike in interactions with the mail server. When the mail server's message log activity exceeds specific levels, PureMessage generates a report.

The `pmx-mlog-watch` utility is run as a scheduled job that monitors the `message_log` for anomalous activity. The thresholds at which actions are triggered are set on the **Local Services: Perimeter Protection Options** page. If anomalies are detected, a report is generated that describes the activity and the envelope sender or relay that was the cause. Alternatively, it can be piped into another program such as `pmx-mlog-react`, which creates entries in the Blacklisted Hosts and Blacklisted Senders lists.

To set the log watch options:

1. Change the **Log Watch Options** text boxes as required.

The text boxes are:

- **Scan Window (min):** The time frame (in minutes) during which PureMessage scans the log for anomalies. For example, if you accept the default of 30 minutes, PureMessage will go back and scan the log to see if any of the conditions in the Log Watch Options have been exceeded within a half-hour period.
- **Max Lines:** The maximum number of lines to be scanned at one time (each line corresponds to one message). This prevents the job from running too long if a lot of messages were received and the number specified for the Scan Window is large. If the number of lines is met or exceeded, a warning is written to the log that is specified in the `log_to` setting in the `pmx.conf` configuration file (by default, `pmx_log`). The default is 10,000 lines.

- **Max Recipients:** The maximum number of recipients a sending relay can specify in one SMTP transaction. If this number is met or exceeded, the "Recipients" counter is incremented for the relay. The default is 50 recipients.
 - **Max Message Size (MB):** If a sender sends a message that reaches or exceeds this value, the "Message Size" counter is incremented for the sender. The default is 10 MB.
 - **Relays:** The number of messages that can be received from one relay during the time period specified in the **Scan Window** text box. If a relay sends more than this number of messages, a report is generated. The default is 5,000 messages.
 - **Senders:** The maximum number of messages that can be received from one sender during the time period specified in the **Scan Window** text box. If a sender sends more than this number of messages, a report is generated. The default is 5,000 messages.
 - **Recipients:** The maximum number of recipients that can be registered by the counter during the time period specified in the **Scan Window** text box. If the recipients counter is triggered more than the specified number of times in the specified time period, a report is generated. The default is 5,000 triggers.
 - **Message Size:** The maximum number of messages that can exceed the specified message size during the time period specified in the **Scan Window** text box. If the message size counter is triggered more than the specified number of times in the specified time period, a report is generated. The default is 5,000 triggers.
2. Once you have finished making changes, click **Save**.
- The settings are saved in `/opt/pmx6/etc/logwatch.conf`

Related tasks

[Managing Scheduled Jobs](#) (page 172)

Related information

[logwatch.conf](#)

[pmx-mlog-watch](#)

[pmx-mlog-react](#)

Enabling or Disabling MTA IP Blocking

MTA-level IP blocking rejects messages originating from IP addresses contained in [SophosLabs](#) block lists and custom block lists. Enabling this option is recommended; it improves performance by blocking spam before it reaches more complex tests in the policy.

Important

Whether you choose to block IP addresses by enabling MTA-level IP blocking or by using the PureMessage policy, PureMessage requires that the IP Blocker Service be enabled. This service is enabled by default. If you opt to block IP addresses using only the PureMessage policy, enabling the `block_dynamic` option described on the `blocklist.conf` man page will cause the additional tests to occur earlier in policy processing, thus improving efficiency.

With `block_dynamic` enabled, PureMessage rejects messages that are sent "Direct-to-MX," a method spammers sometimes use to bypass the sending MTA (and any intermediate MTAs), and send messages directly to the machines hosting the MX records for the intended recipients.

This makes it possible to block spam from hosts that have not yet established a reputation, but are very likely to be sending spam. These additional checks, which make use of the Sophos Sender Genotype, are referred to as proactive protection control because they allow PureMessage to reject connections from servers with dynamic IP addresses.

For an explanation of Sophos Labs IP address classifications, see the [Sophos website](#).

The `block_dynamic` option can only be enabled from the command line. See the `blocklist.conf` man page for more information.

Messages are blocked based on the latest data from Sophos Labs, and any IP addresses or fully qualified hostnames that have been specified in the **IP Blocking Exception List** and **IP Blocking Exclusion List**. For more about these lists, see “About PureMessage Default Lists” in the *Manager Reference*.

The **Local Services: MTA IP Blocking** page of the **Local Services** tab allows you to enable/disable IP blocking.

Note

MTA-level IP blocking must be enabled or disabled manually on each server in multi-server deployments (not on the Central Server Manager).

To set MTA IP blocking:

1. On the **MTA IP Blocking** page of the **Local Services** tab, select the **Enable** check box.
2. You are prompted to restart both your mail transfer agent (MTA) and the Scheduler Service. Click the **Restart now** buttons next to each of these prompts.

Note

- If you want to configure IP blocking with an external or third party version of sendmail or Postfix, manual steps are required. See the appropriate “Configuring IP Blocking” section in the *Getting Started Guide* for more information.
- If you want to authenticate connections using SMTP-AUTH while MTA-level blocking is enabled, you must modify PureMessage Postfix. For instructions, see “Configuring SMTP Authentication with the MTA IP Blocker” in the Sophos Knowledgebase. SMTP-AUTH is not supported for external Postfix installations nor for any type of sendmail installation.

Related concepts

[About PureMessage Default Lists](#) (page 118)

Related tasks

[Configuring IP Blocking \(External Sendmail Version\)](#) (page 48)

[Configuring IP Blocking \(External Postfix Version\)](#) (page 51)

Related information

[pmx-blocker](#)

[pmx-blocklist](#)

[blocklist.conf](#)

[Configuring SMTP Authentication with the MTA IP Blocker](#)

Setting Auto Refresh

1. From the drop-down list in the **Auto Refresh** section of the sidebar, select the desired refresh period.
2. Click **OK**.

The status of the background services and scheduled jobs that are listed on this page are refreshed according to the selected frequency.

2.7 Server Groups Tab

The **Server Groups** tab lets you manage services in a multi-server PureMessage deployment from the [Central Server Manager](#) (CSM) server. You can also publish policy and related configuration settings from the CSM server to other PureMessage hosts. Server groups allow you to do the following:

- View the status of all PureMessage servers on the network.
- Add and remove edge servers.
- Start and stop PureMessage services on edge servers.
- Build and test configurations on the central server, and then distribute configurations to edge servers.
- Generate reports for edge servers (or groups of edge servers) from the central server.

Edge servers are not "aware" of the central server, or of their status as edge servers. Therefore, it is possible to manually configure an edge server and then accidentally overwrite that configuration by updating the edge server from the central server. One way to prevent this is by limiting user permissions on edge servers. See the "Managing the HTTPD (Manager) Service" section of the *Administrator's Reference* for information about configuring administrator and administrator group accounts.

Edge servers can only be accessed when the HTTPD (Manager) service is running.

Every PureMessage installation has a default "RPC User" account configured for the Manager service. This account has a limited set of privileges, sufficient for starting and stopping services and updating configuration files.

Scan the Network or Find Hosts

After a new installation, the **Server Groups** tab appears empty except for a message that there are "No hosts or groups configured" and a link to "Scan the network".

To scan your network to automatically populate the **Network Status** table, either click **Scan the Network** in the content pane or **Find Hosts** on the sidebar of the **Server Groups** tab. The **Find Hosts** function sends a broadcast UDP query to port 18080 of all hosts on the network. If the PureMessage Manager is running, it responds, and a remote host entry is added to the host where **Find Hosts** was run. The new host is displayed on the Network Status (default) page.

Note

When scanning for hosts, PureMessage only detects hosts on which SSL is enabled.

To stop or restart services running on remote hosts, you must manually configure authentication information for the remote host.

Related concepts

[Managing the HTTPD \(Manager\) Service](#) (page 154)

Related information

[servergroups.conf](#)

2.7.1 Managing Hosts

You must set the password for the “rpcuser” account in Local Services : HTTPD (Manager) : RPC User on each remote host that you want to access before you can fully configure Server Group hosts.

Hosts are PureMessage servers located on the same network as the current server. Hosts are added to the Network Status table, making it possible to distribute publications to, manage services on, and access the PureMessage Manager on remote hosts.

Adding a Host

Hosts can be added automatically by clicking **Scan the network** on the **Server Groups** tab or **Find Hosts** on the sidebar; however hosts added this way don't have all the information that is required to remotely access the host, so the host information has to be edited.

To add a host:

1. On the sidebar of the **Server Groups** tab, click **Add Host**.
The **Add Host** page is displayed.
2. In the **Host Alias** text box of the **Host** form, enter a meaningful name for the host.
3. In the **Description** text box, enter a description of the host's role.
4. In the **Hostname** text box, enter the fully qualified domain name (for example, "mailserver.myco.com") of the host.
5. In the **Port** text box, enter "18080" (the port on which the PureMessage Manager runs).
6. In the **User** text box, enter "rpcuser".
7. In the **Password** text box, enter the rpcuser password as set in the preparation for this procedure.
8. Click **Save**.

Editing a Host

To perform any Server Group operations on a remote host, you must log in to an account that is authorized to manage the affected services. Every PureMessage installation has a default "rpcuser" account, which is configured with appropriate access rights for accessing remote servers.

Hosts for which authentication information has been configured have a key icon displayed in the first (leftmost) cell of the hosts tables on the **Server Groups: Network Status** page. Hosts without this icon (often hosts added by using the **Scan the network** or **Find Hosts** features) must have this information added before they can be used for any of the Server Group remote management operations.

To edit a host:

1. In a hosts table on the **Server Groups** tab, click the name of the host that you want to edit.
The **Edit Host** page is displayed.
2. Modify or fill in the text boxes that you want to update.
3. Below the table, click **Save**.
An information dialog box appears, informing you of the result of the operation.
4. Click **Continue**.
You are returned to the **Network Status** page.

Managing Services on Remote Hosts

You must set the password for the "rpcuser" account on the **Local Services > HTTPD (Manager) > RPC User** page of each remote host that you want to access before you can fully manage services remotely.

The hosts tables on the Server Groups: Network Status page display the status of three services for each host: Milter (the PureMessage mail filter), SMTP (the mail transfer agent), and HTTP (the Manager web server). Each status indicator (for example, "unknown", "stopped" or "running") is a link to a page, **Server Groups : Status of <servicename> service on <host>:18080**, that displays information about that service, as well as controls used to manage that service.

To start, restart, or stop a service on a remote host:

1. Click the service status descriptor ("unknown", "stopped" or "running") in one of the hosts tables on the **Network Status** page.

The **Server Groups: Status of <servicename> service on <host>:18080** page for the selected service on the selected host is displayed.

2. Click the control for the operation that you want to perform on this service, **Send Start Request**, **Send Restart Request** or **Send Stop Request**.

An information message is displayed, stating that the request is being processed.

3. Click Refresh.

You are returned to the Server Groups: Status of <servicename> service on <host>:18080 page.

Logging in to a Remote Host

Once a remote host is fully configured and authorization data has been entered, you can also open the PureMessage Manager on that host

To log in to a remote host:

1. In the hosts table, on the **Network Status** page, click either the name of the host that you want to log in to, or the service status descriptor for any service running on that host.

Depending on whether you clicked on the hostname or a service status descriptor, either the **Edit Host** page or the **Status of <servicename> service on <host>:18080** page is displayed.

2. Click **Remote login**.

A new browser window opens displaying the **Dashboard** tab of the PureMessage Manager for the remote host.

Note

Only those pages that this user account is authorized to view are displayed.

3. Perform whatever operations you want in the remote host's Manager. Log out and close the new browser window when you are done.

Deleting a Host

1. In a hosts table on the **Server Groups** tab, click the name of the host that you want to delete.

The **Edit Host** page is displayed.

2. Below the table, click **Delete**.

A confirmation dialog box appears, asking you to confirm or cancel the delete operation.

3. Click **Delete**.

An information dialog box appears, informing you of the result of the operation.

4. Click **Continue**.

You are returned to the **Network Status** page.

2.7.2 Managing Groups

Host Groups are collections of remote hosts created to act as the basis for generating reports or as the "recipient list" for publishing configuration data, although reports can also be run for individual hosts.

After running the **Scan the network** or **Find Hosts** functions the **Network Status** page initially displays the newly found hosts in the **Unsorted hosts** table.

Creating a Group

1. On the sidebar of the **Server Groups: Network Status** page, click **Add Group**.

The **Create Group** page is displayed.

2. In the **Create** table, enter a group name in the **Group** text box and a description in the **Description** text box.
3. Click **Save**.

A message box is displayed, informing you of the result of the operation.

4. Click **Continue**.

You are returned to the **Network Status** page. The group that you created appears below the **Unsorted hosts** table.

Related tasks

[Adding Hosts to a Group](#) (page 183)

[Deleting a Group](#) (page 184)

Adding Hosts to a Group

Newly created host groups do not contain any hosts.

To add hosts to a group:

1. On the **Server Groups: Network Status** page, in the **Host** column, click the hostname of a host that you want to add to a group.

The **Edit Host** page is displayed.

2. Select the check box beside the group name to which you want the host added, and click **Save**.

A single host can belong to more than one group.

A message box is displayed, informing you of the result of the operation.

3. Click **Continue**.

You are returned to the **Network Status** page. The selected host is now displayed in the appropriate group table.

Note

You can subsequently modify the group information by clicking the group name on the first line of the group table.

4. Repeat steps 1 through 3 for each host that you want to add to the available group(s).

Generating a Group Report

1. On the **Reports** tab, on the central server (the PureMessage system with the CSM role installed on it), click **Reports** on the sidebar.

The **Reports** page is displayed.

2. Click the name of the report that you want to view.

A report-specific page is displayed, with the selected report displayed in the content pane.

3. In the **Modify Report** form in the sidebar, select the edge server or server group for which you want the report to cover, and click **Change**.

The report on the selected edge server or server group is displayed.

Optionally, **Export** or **Schedule** the report.

Related tasks

[Exporting Reports](#) (page 152)

[Scheduling Reports](#) (page 152)

Deleting a Group

1. On the **Server Groups: Network Status** page, click the group name on the first line of the group table that you want to delete.

The **Edit Group** page is displayed.

2. Below the table, click **Delete**.

A confirmation dialog box appears, asking you to confirm or cancel the delete operation.

3. Click **Yes Delete**.

An information dialog box appears, informing you of the result of the operation.

4. Click **Continue**.

You are returned to the Network Status page, where the group you selected for deletion no longer appears, and any hosts included in only that group are returned to the **Unsorted hosts** table.

2.7.3 Managing Publications

Publications are sets of configuration files on the central server that are distributed to remote PureMessage hosts (edge servers). Edge servers are subscribed to these publications on the central server. Multiple publications can be configured, each listing one or more configuration files. Edge servers can be subscribed to one or more publications. Existing publications are listed under the Publications heading on the sidebar of the Server Groups tab.

When PureMessage is installed, default publications are created that include configuration files associated with each major area of PureMessage functionality. For more information about these

configuration files, refer to “Default Publications” in the Server Groups Management section of the *Administrator's Reference*.

Settings that can be configured by end users via the End User Web Interface (EUWI) *are not* distributed via publications. The `enduser_ui.conf` file contains general configuration data for the EUWI, rather than specific preferences configured by individual users.

Note

You must set the password for the `rpcuser` account on the Local Services: HTTPD (Manager): RPC User page of each remote host that you want to access before you can fully configure server group hosts.

Related concepts

[Default Publications](#) (page 199)

Creating Publications

1. On the **Server Groups** tab's sidebar, beside **Publications**, click **New [+]**.

The **Create New Publication** page is displayed.

2. In the **Publication Name** text box, enter a meaningful name for the publication.
3. In the **Description** text box, enter a description of the publication's purpose or content. For example, "configuration for synchronizing east coast MTAs".
4. Click **Save**.

The **Edit Publication** page is displayed, with creation date shown in the bottom row of the **Properties** form. A warning is displayed, indicating that "There are no shared files, lists or maps in this publication."

Note

Shared resources cannot be added using the Manager. If you choose to add shared resources to a new publication, you must complete this procedure, and then use the PureMessage command-line program `bin/pmx-share` and the configuration file `etc/publications.conf` to add the configuration resources that you want to share in the new publication.

There is also a warning that "This publication doesn't have any subscribers."

5. Add resources (configuration files) to the publication:

- a) On the **Policy** tab, open the policy, list, or map that you want to add to the publication for editing.

The policy, list, or map must not be configured to be shared in an existing publication. If it is, you must remove it from its current publication before you can add it to another publication. The publication status of the policy, list, or map is indicated in a box at the top of the content pane.

- b) In the **You can publish this configuration to other hosts** box at the top of the content pane for the policy, list, or map that you want to add to the publication, select the publication from the relevant the drop-down list, and click **Share**.

The **Edit Publication** page is redisplayed, showing the added policy, list, or map in the **Published Objects** table.

6. In the drop-down list below the “no subscribers” warning, select the host that you want to configure as a subscriber, and click **Subscribe**.

The **Edit Publication** page is redisplayed, showing the added subscriber hosts in the **Subscribers** form.

7. Repeat the previous step as often as required to build the list of subscriber hosts that you want to receive the publication described in the Properties form.

Related information

[pmx-share](#)

Modifying Publications

Publications can be modified by removing resources (configuration data) or subscribers (hosts that are updated with the current server's configuration data).

Note

Resources can only be added to publications by using the PureMessage command-line program `bin/pmx-share` and the configuration file `etc/publications.conf`.

To modify a publication:

1. In the **Publications** section of the **Server Groups** sidebar, click the name of the publication that you want to modify.

The **Edit Host** page is displayed.

2. You can modify the **Publication Name** or the **Description** by editing those text boxes in the **Properties** form and clicking **Save**.

The "Date Modified" line at the bottom of the **Properties** form is updated, indicating that the publication has been successfully saved and, if the **Publication Name** was changed, the name of the publication is updated in the sidebar.

3. You can remove resources by selecting the check box beside the resource in the **Published Objects** table and clicking **Revoke**.

The **Edit Host** page is redisplayed with the selected resource removed from the **Published Objects** table.

4. You can add subscribers by selecting a host from the **select host to subscribe** drop-down list and clicking **Subscribe**.

The **Edit Host** page is redisplayed with the selected host added to the **Subscribers** table.

5. You can remove subscribed hosts from the **Subscribers** table by selecting the check box beside the subscribed host that you want to remove and clicking **Unsubscribe**.

The **Edit Host** page is redisplayed with the selected host removed from the **Subscribers** table.

Synchronizing Publications

To synchronize a publication (push it out to the subscribed hosts):

1. In the **Publications** section of the **Server Groups** tab sidebar, click the name of the publication that you want to synchronize.

The **Edit Publication** page is displayed.

2. Ensure that the **Published Objects** and the **Subscribers** lists contain all of the items that you want.

3. In the **Subscribers** table, select the check boxes beside all of the hosts that you want synchronized to the select publication, and click **Synchronize**.

The **Date Modified** text box beside each of the hosts that you selected is filled in or updated, indicating that the configuration synchronization was successful.

Note

The distribution of publications does not automatically restart the affected services on the edge server(s). These services must be restarted manually.

Related tasks

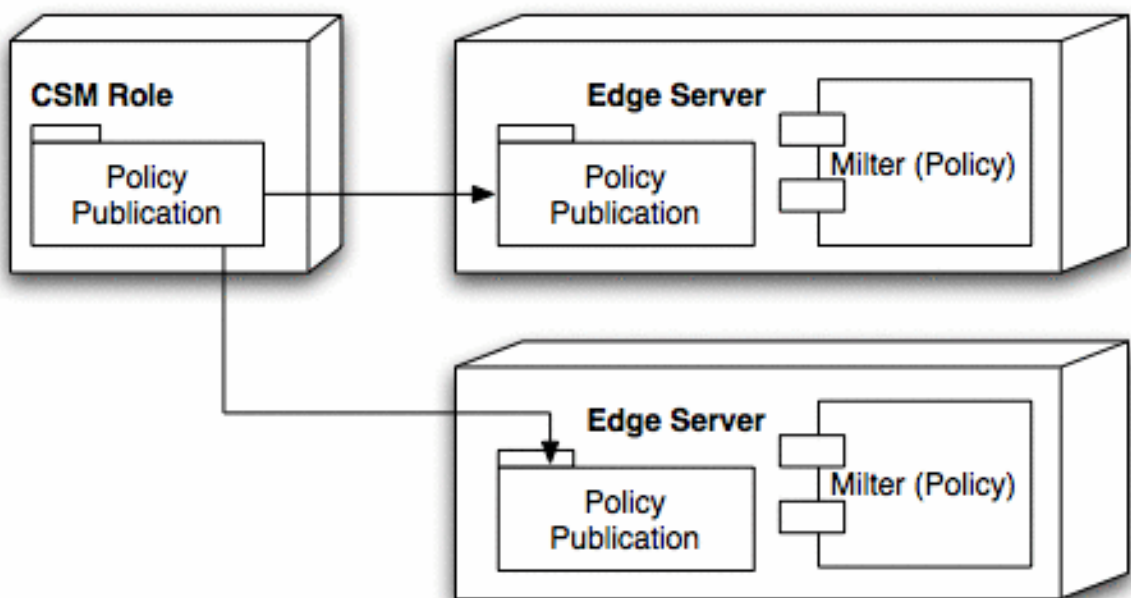
[Managing Services on Remote Hosts](#) (page 182)

Related information

[pmx-share](#)

Configuring and Distributing Policy Settings

Example: Change the policy script on the central server and migrate changes to all edge servers.



1. Modify the policy on the central server: Publications are displayed on the sidebar of the **Server Groups** tab. Click the default "Policy Publication" to make the desired changes.
2. Distribute policy publication changes to each edge server: On the **Server Groups** tab, select the **Policy** publication on the sidebar. Hosts and groups subscribed to the selected publication are displayed in the **Subscribers** table at the bottom of the page. Select the check box beside the host(s) or group(s) to update, and then click **Synchronize**.
3. Restart the Milter (Policy) service on each edge server: Edge servers must be restarted after publications are updated for the changes to take effect. On the **Server Groups** tab, click the **Milter running** link beside the edge server that requires restarting. The milter service status for that edge server is displayed. Click **Send Restart Request**. A restart signal is sent to that server. Restart the milter process for each edge server in the PureMessage configuration. From the command line, use the `pmx-milter` command to restart the milter process on each edge server.

Related concepts

[Server Groups Tab](#) (page 180)

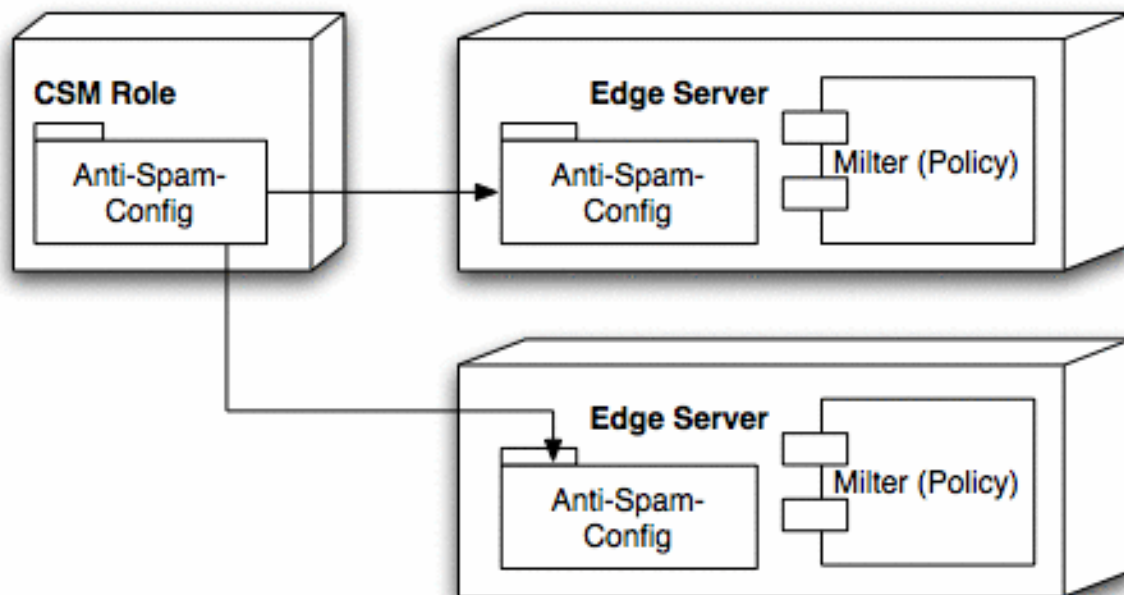
[Default Publications](#) (page 199)

Related information

[pmx-milter](#)

Configuring and Distributing Anti-Spam Settings

Example: Alter the anti-spam configuration on the central server and distribute the modified files via the Anti-Spam-Config publication to all edge servers.



1. Modify the anti-spam configuration on the central server: Publications are displayed on the sidebar of the **Server Groups** tab. Modify the desired weights, probabilities, or custom rules in the anti-spam configuration on the **Anti-Spam Rules** page, or by using the `pmx-spam` command-line program.
2. Distribute modified files via Anti-Spam-Config to each edge server: On the **Server Groups** tab, click the **Anti-Spam-Config** publication on the sidebar. Hosts and groups subscribed to the selected publication are displayed in the **Subscribers** table at the bottom of the page. Select the check box beside the host(s) or group(s) to update, and then click **Synchronize**.
3. Restart the Milter (Policy) service on each edge server: Edge servers must be restarted after publications are updated for the changes to take effect. On the **Server Groups** tab, click the **Milter running** link beside the edge server that requires restarting. The milter service status for that edge server is displayed. Click **Send Restart Request**. A restart signal is sent to that server. Restart the milter process for each edge server in the PureMessage configuration. From the command line, use the `pmx-milter` command to restart the milter process on each edge server.

Related concepts

[Server Groups Tab](#) (page 180)

[Default Publications](#) (page 199)

Related information

[pmx-milter](#)

Deleting Publications

You can delete unwanted publications from the sidebar.

To delete a publication:

1. In the **Publications** section of the **Server Groups** tab sidebar, click the name of the publication that you want to delete.

The **Edit Publication** page is displayed.

2. Click **Delete** at the bottom of the **Properties** form.

A message box is displayed, informing you of the deletion.

3. Click **Continue**.

You are returned to the **Network Status** page and the selected publication is removed from the sidebar.

2.8 Support Tab

The **Support** tab provides access to a variety of PureMessage support and system maintenance features.

The **Licensed Components** (default) page displays the PureMessage components installed on the system. The **Licensed Components** table indicates the name, license type, issue date, and expiration date for each component.

In addition to links to other PureMessage support features and services, the sidebar provides convenient links to the PureMessage website, the mailing list archive, and the *PureMessage User Guide*.

Click the **Check for Updates** link on the sidebar to access a page that lets you query for available software updates. If any updates are available, you can retrieve them by way of the PureMessage Installer.

Click **View Installed Packages** on the sidebar to see which PureMessage Core Packages and Support Packages are installed. The tables on this page indicate the version number and description of each package.

The **Available Updates** page displays the **Package Repositories** drop-down list. Select a repository from the list and click **Query** to get more information. A list of all installed packages with their version numbers and their update status is displayed. Run `pmx-setup` from the command line to install all of the available updates.

2.8.1 Managing Package Repositories

Package repositories are URLs to sites from which PureMessage system software updates can be upgraded and installed. The **Package Repositories** page displays any previously configured PureMessage package repositories.

Relocating Repositories

To change a repository location:

1. In the **Update Software** section of the sidebar, click **Package Repositories**.

The **Package Repositories** page is displayed.

2. In the **Available Repositories** table, click the name of the repository that you want to relocate.

The **Edit Repository** page is displayed.

3. In the **Name** text box, change the display name for the repository, if required.
4. In the **Location** text box, change the URL for the repository, if required.
5. Click **Save**.

You are returned to the **Package Repositories** page, and the changes are displayed in the table.

Adding a Repository

1. Below the **Available Repositories** table, click **Add**.

The **Add Repository** page is displayed.

2. In the **Name** text box, enter a display name for the repository.
3. In the **Location** text box, enter the URL for the repository.
4. Click **Save**.

You are returned to the **Package Repositories** page, and the added repository is displayed in the table.

Deleting a Repository

1. In the **Available Repositories** table, select the check box beside the repository that you want to delete.
2. Click **Delete**.

The selected repository is removed from the table.

2.8.2 Sending a Support Request

A PureMessage Support Request is an emailed request for support that attaches all of the configuration and log file data a Sophos support engineer should require to address your issue. The generation and sending of the request is partially automated.

To email a request to Sophos support:

1. In the **Other Tasks** section of the sidebar, click **Send Support Request**.

The **Send Support Request** page is displayed.

2. In the **Email Request Information** table, enter the required From information in the Your Name, E-mail address, and Company Name text boxes.
3. Modify the **Subject** to accurately describe your issue.
4. Select the **Request type** from the drop-down list.
5. Select the **Component** from the drop-down list.
6. Ensure that all of the relevant attachments are selected. Each attachment type has an associated check box. All check boxes are selected by default. You can click **view** next to each attachment type to see what information will be sent.
7. Enter any information that you think will add to the understanding of the issue in the **Additional Information** text box.
8. Click **Send**.

You will receive warnings if there are any problems. If not, a message is displayed informing you that the request was successfully sent.

2.8.3 Sharing Data with Sophos

When this option is enabled, statistical reports are sent to Sophos every five minutes. These reports contain information about the status of your system, the Sophos version, and key mail traffic statistics. These reports, which are based upon log file data, do not contain actual message content, or content that identifies specific mail users.

Version Information

Each report contains the following entries:

- Report-Version: The version of the report format.
- Report-TimeZone: The system time zone in +-hhmm format, indicating the relationship to Greenwich mean time (GMT).
- Report-Time: The GMT time at which the report was generated.
- Report-Last-Run-Time: The GMT time at which the last run began.
- First-Message-Scanned: The local time of the first message included in the report.
- Last-Message-Scanned: The local time of the last message included in the report.
- System-Product: The product for which the report is generated (in this case, PureMessage for Unix).
- System-Version: The version number of the product.
- System-SerialNo: The PureMessage serial number associated with the installation.
- System-Id: A string uniquely identifying this installation of the product. The string consists of the product serial number and the location ID.
- Version-PureMessage-AntiSpam-Data: The version of the AntiSpam-Data package.
- Version-PureMessage-AntiSpam-Engine: The version of the AntiSpam-Engine package.
- Version-PureMessage-AntiSpam-Utils: The version of the AntiSpam-Utils package.
- Version-PureMessage-Sophos-Data: The version of the Sophos-Data package.
- Version-PureMessage-Sophos-Engine: The version of the Sophos-Engine package.
- Version-PureMessage-Sophos-SAVI: The version of the Sophos-savi package.
- Version-PureMessage-Blocklist: The version of the Blocklist package.
- Version-PureMessage-Blocklist-Daemon: The version of the Blocklist-Daemon package.
- Version-PureMessage-Blocklist-Data: The version of the Blocklist-Data package.
- Blocker-Status: The status of the MTA-level IP blocker. This is displayed as disabled, enabled or not installed. If IP blocking is enabled, this entry will also indicate whether dynamic and HELO checks have been enabled as well.

Sender IP Information

Each report also contains a one-line entry for each sender IP address. Each line indicates the number of messages that match each of the following criteria:

- IPBlocker: total connections
- IPBlocker: connections rejected
- IPBlocker: connections accepted

- MTA: total messages
- total messages scanned for spam
- messages detected as spam
- total messages scanned for viruses
- total messages with virus detected
- total messages with suspicious attachments
- virus names detected in e-mail from this IP

[SophosLabs](#) spam analysts can use this mail traffic data to compile statistics about sender reputation, and create more comprehensive block lists.

SXL Information

The reports also include SXL-related data. SXL is the infrastructure that Sophos uses to submit real-time, DNS-based queries to SophosLabs regarding IP addresses, URIs within messages, and image fingerprints. Queries are triggered when the anti-spam engine has been unable to determine if a message is spam. These real-time lookups are enabled by default and provide minimal latency between the time that Sophos makes new anti-spam data available and when it is available for use by the anti-spam engine. The following data is included in each statistical report:

- frequency of anti-spam rule hits in messages that were determined to be spam or not spam
- number of messages processed
- number of messages determined to be spam before and after SXL queries
- number of messages that didn't generate SXL queries because there was no server response
- total CPU time as compared to the time spent waiting for DNS responses.
- frequency of the various types of SXL queries (IP, URI, checksum, etc)
- per-server statistics on latency, query count, query sizes, timeouts, and errors for Sophos SXL servers receiving and sending data

Enabling Data Sharing

1. On the **Support** tab sidebar, click **Share data with Sophos**.
2. Select the **Share Data with Sophos** check box.
3. Click **Save**.

3 Administrator's Reference

The Administrator's Reference provides an in-depth examination of PureMessage. PureMessage began as a Unix-based, command-line mail filtering tool, and it still retains many aspects of its origins.

This guide has three purposes:

- It stands as an introduction for Unix administrators who prefer to work from the command line.
- It is a guide to the advanced configuration and management of PureMessage, which often must be done at the command line.
- It provides troubleshooting techniques that require you to work at the command line.

The *Administrator's Reference* assumes that you have a basic familiarity with Unix, including the Unix file system, Unix editors, and the general workings of Unix-based client-server applications.

Although this reference contains detailed explanations of the command line applications, configuration files, and log files associated with PureMessage, you can often find greater detail in the PureMessage man pages, which are accessed using the `man` command as the `pmx6` user in the form:

```
man <program_name>
```

Where `<program_name>` is the name of the program whose documentation you want to view.

3.1 PureMessage Utilities

This section describes the general PureMessage utilities, which provide installation, startup, general configuration, update, and uninstall capabilities.

PureMessage command-line applications and configuration files can be used to install, start, stop, set general configuration options, and update or remove components. This is of interest to those who prefer working from the command line, and it is important for troubleshooting. For full documentation of these tasks as procedures, or when working with the PureMessage graphical user interface (the PureMessage Manager), please see the *PureMessage Getting Started Guide* and the *PureMessage Manager Help*.

3.1.1 Install, Startup, Shutdown

This page describes the PureMessage command-line utilities that display the license, run the installer, verify the installation, start and stop PureMessage, and test the basic configuration of PureMessage.

- `/opt/pmx6/bin/pmx-license` : Displays the Copyright and Trademarks acknowledgements for PureMessage.
- `/opt/pmx6/bin/pmx-wfetch` : A simple HTTP download client used to download the PureMessage installation packages.
- `/opt/pmx6/bin/pmx-setup` : The PureMessage installer application.
- `/opt/pmx6/bin/pmx-verify` : A utility that checks the integrity of a PureMessage installation.
- `/opt/pmx6/bin/pmx-init` : A system startup wrapper for the PureMessage control program.
- `/opt/pmx6/bin/pmx` : The PureMessage control program used to start, restart, stop or get the status of all PureMessage Services.

- `/opt/pmx6/bin/pmx-test` : A utility that verifies a PureMessage installation's functionality by sending test messages, including spam and pseudo-virus test messages.

Related Configuration Files

- `/opt/pmx6/etc/pmx.conf` : The main PureMessage configuration file, with many configuration settings.

Related information

[pmx-license](#)

[pmx-wfetch](#)

[pmx-setup](#)

[pmx-verify](#)

[pmx-init](#)

[pmx](#)

[pmx-test](#)

[pmx.conf](#)

3.1.2 Configuration Utilities

This page describes the PureMessage command-line configuration viewer and editor as well as a utility that sources the PureMessage user's environment when running applications as scheduled jobs.

- `/opt/pmx6/bin/pmx-config` : An important utility used to view or edit PureMessage configuration (.conf) files.
- `/opt/pmx6/bin/pmx-env` : Used to source the PureMessage user's environment if you want to run PureMessage applications as scheduled jobs.

Related information

[pmx-config](#)

[pmx-env](#)

3.1.3 Update, Uninstall and Relocate

This page describes the PureMessage package and anti-virus data update utilities, the PureMessage component uninstaller, and the utility for relocating a PureMessage installation.

- `/opt/pmx6/bin/pmx-sophos-data` : Updates the Sophos Anti-Virus data files.
- `/opt/pmx6/bin/pmx-update-all` : Deprecated. Updates all installed PureMessage packages.
- `/opt/pmx6/bin/pmx-remove` : Used to uninstall specific PureMessage components.
- `/opt/pmx6/bin/pmx-location` : Used to change the configured IP address and hostname of a PureMessage server.

Related information

[pmx-sophos-data](#)

[pmx-update-all](#)

[pmx-remove](#)

[pmx-location](#)

3.2 PureMessage Services

This section describes the management of the services that provide PureMessage functionality. All of this functionality is available via the PureMessage Manager, so the command-line alternatives are presented for those who prefer to work from the command line and for troubleshooting.

PureMessage functionality depends on the constant operation of the following services:

3.2.1 PureMessage Manager

The PureMessage Manager provides a graphical user interface for managing PureMessage.

- `/opt/pmx6/etc/init.d/pmx-manager` : The PureMessage Manager service control program.
- `/opt/pmx6/bin/pmx-manager-status` : Used to print the PureMessage Manager status information.
- `/opt/pmx6/bin/pmx-manager-config` : Used to print the PureMessage Manager configuration options.
- `/opt/pmx6/bin/pmx-manager-passwd` : Manages the PureMessage Manager user database.
- `/opt/pmx6/bin/pmx-cert` : Generates a self-signed certificate for Manager access.

Related Configuration Files

- `/opt/pmx6/etc/utf8.conf` : Allows you to enable support for UTF-8 encoding of East Asian languages. This support affects the Quarantine Search Results page and the Quarantine Message Details page in the PureMessage Manager.

Related information

[pmx-manager](#)
[pmx-manager-status](#)
[pmx-manager-config](#)
[pmx-manager-passwd](#)
[pmx-cert](#)
[utf8.conf](#)

3.2.2 Mail Filter Service

The Mail Filter service provides the main functionality of PureMessage: the filtering of viruses, spam and policy violations in the email handled by the mail transfer agent.

- `/opt/pmx6/etc/init.d/pmx-milter` : The PureMessage milter control program.
- `/opt/pmx6/bin/pmx-milter-add` : Adds mail filter (milter) sections to the `pmx.conf` file.

Related information

[pmx-milter](#)
[pmx-milter-add](#)

3.2.3 IP Blocker Service

The IP Blocker service rejects messages originating from IP addresses blacklisted by Sophos Labs™. Enabling this option can improve performance by blocking spam before it reaches more complex tests in the Policy.

Note

If you want to enable MTA-level IP blocking, but you require that a select group of internal email addresses be exempt from blocking, you can create a recipient exception list.

Be aware, however, that doing so will result in a higher rate of spam because any message addressed to multiple recipients that contains at least one address from the recipients exception list will be delivered to all recipients of that particular message, even if the recipients are not on the exception list. For details about configuring a recipient exception list or an explanation of your options with regard to this feature, contact Sophos support.

- `/opt/pmx6/etc/init.d/pmx-blocker` : The IP Blocker service control program.
- `/opt/pmx6/bin/pmx-blocklist` : Performs conversions on blocklist data.
- `/opt/pmx6/etc/pmx.d/blocklist.conf` : Used to set configuration options for the MTA-level IP Blocker.

Related information

[pmx-blocker](#)
[pmx-blocklist](#)
[blocklist.conf](#)
[Sophos support](#)

3.2.4 PureMessage Database

The PureMessage database stores metadata about PureMessage data, including information on quarantined messages, lists and maps, end user data and reports data.

- `/opt/pmx6/bin/pmx-postgres-enable` : A utility for PureMessage PostgreSQL database server setup.
- `/opt/pmx6/bin/pmx-resources-init` : Initializes the resources system.
- `/opt/pmx6/bin/pmx-database` : The service control program for the PostgreSQL quarantine database server.
- `/opt/pmx6/bin/pmx-schema` : Used to initialize and display the PMX database schema.
- `/opt/pmx6/bin/pmx-pg-autovac` : A utility for maintaining PostgreSQL database tables in a persistent state. This tool is for use with PostgreSQL 7.4 only.
- `/opt/pmx6/bin/pmx-profile` : Synchronizes resources between local and database stores.

Related information

[pmx-postgres-enable](#)
[pmx-resources-init](#)
[pmx-database](#)
[pmx-schema](#)
[pmx-pg-autovac](#)

[pmx-profile](#)

3.2.5 End User Web Interface

The PureMessage End User Web Interface (EUWI) allows end users access to any of their quarantined messages. This service can be enabled or disabled, and a range of capabilities can be set, allowing end users very limited or more extensive options.

- `/opt/pmx6/bin/pmx-httpd` : The End User Web Interface service control program.
- `/opt/pmx6/bin/pmx-rpc-enduser` : Used to query and test the End User Web Interface.

Related Configuration Files

- `/opt/pmx6/etc/enduser/auth.d/ldap.conf` : Configures the interface for authenticating the end user via LDAP.
- `/opt/pmx6/etc/utf8.conf` : Allows you to enable support for UTF-8 encoding of East Asian languages. This support affects the Quarantine Search Results page and the Quarantine Message Details page in the End User Web Interface.

Related information

[pmx-httpd](#)[pmx-rpc-enduser](#)[ldap.conf](#)[utf8.conf](#)

3.2.6 PureMessage Scheduler

The PureMessage Scheduler service runs specific commands at specified times and intervals. Several jobs are required to enable the full set of PureMessage features. This service lets you stop and start such jobs using the `pmx stop` and `pmx start` commands.

- `/opt/pmx6/etc/init.d/pmx-scheduler` : Sets which scripts are run by the PureMessage scheduler daemon and when they are run.

Related Configuration Files

- `/opt/pmx6/etc/scheduler.conf` : Contains the list of jobs handled by the Scheduler.

Related information

[pmx-scheduler](#)[scheduler.conf](#)

Scheduled Jobs Syntax

The PureMessage Scheduler is a cron-like service that runs PureMessage commands at scheduled times. The advantage of using the Scheduler instead of cron is that it allows the scheduled jobs to be easily stopped and started along with PureMessage itself or on command.

On the **Local Services** tab, click **Scheduler Service** to view the service status or to start and stop the Scheduler. To edit individual services, click the Job Name in the **Scheduled Services** table.

Scheduled jobs can also be added by editing `/opt/pmx6/etc/scheduler.conf`. Each event is described in an `<event>` section. For example:

```
<event queue>
  desc = 'Run the PureMessage Queue'
  enabled = 1
  action = /opt/pmx6/bin/pmx-queue
  type = exec
  #run every five minutes
  <when>
    s = 1
    m = */5
  </when>
</event>
```

The `<when>` block determines how often the Scheduler runs an event. If the `<when>` block does not contain a value or if it is missing, the time defaults to once a minute. For events that run less often, you must specify times and dates. For example, to schedule an event to run at 2:40 a.m. on the 15th and 30th day of each month and every Sunday morning, enter:

```
<when>
  s = 1
  m = 40
  h = 2
  md = 15
  md = 30
  wd = 7
</when>
```

Note

You must include the `s = 1` or it will run at every second of the specified minute (60 seconds, from 2:40 to 2:41).

The Scheduler's command-line options are available via `pmx-scheduler`.

Related concepts

[Managing the Scheduler Service](#) (page 167)

Related tasks

[Managing Scheduled Jobs](#) (page 172)

Related information

[pmx-scheduler](#)

3.3 Server Groups Management

This section describes the command-line programs and configuration files used to set up and manage multi-server PureMessage deployments. Server Groups make it possible to manage multi-server deployments from a single server, the Centralized Server Manager (CSM).

PureMessage Server Groups provide a convenient method for sharing and synchronizing configuration settings between hosts in a multi-server PureMessage deployment. Server Groups functionality is

available via the PureMessage Manager, although a list of the configuration files and other resources included in each publication is shown in the Default Publications section.

- `/opt/pmx6/bin/pmx-service` : An interface for PureMessage services running on remote servers.
- `/opt/pmx6/bin/pmx-share` : Synchronizes PureMessage configuration files with other PureMessage hosts.
- `/opt/pmx6/bin/pmx-resources-init` : Initializes the resources system.
- `/opt/pmx6/bin/pmx-profile` : Synchronizes resources with other PureMessage hosts.

Related Configuration Files

- `/opt/pmx6/etc/servergroups.conf` : Contains the list of PureMessage hosts that are subscribed to configuration publications.
- `/opt/pmx6/etc/publications.conf` : Lists the configuration files that are published to subscribed hosts. This file should be set using the `pmx-share` program.

Related information

[pmx-service](#)

[pmx-share](#)

[pmx-resources-init](#)

[pmx-profile](#)

[servergroups.conf](#)

[publications.conf](#)

3.3.1 Default Publications

This page lists the configuration files and resources that are included in each of the default publications.

PureMessage-Database-Config Publication

The PureMessage-Database-Config publication contains the shared configuration files for PureMessage databases.

- `pmx.d/pmdb.conf` (configuration file)

Anti-Virus-Conf Publication

The Anti-Virus-Conf publication contains general configuration files for anti-virus scanning.

- `virus.conf` (configuration file)
- `templates/<language>/virus.d/cantclean.tmpl` (message template for uncleanable viruses)
- `templates/<language>/virus.d/cantscan.tmpl` (message template for unscannable messages)
- `virus.d/cantscan.conf` (configuration file for unscannable messages)

Sophos-Anti-Virus-Conf Publication

The name of the anti-virus publication is `Sophos-Anti-Virus-Conf`. This publication contains `sophos.conf`, the associated configuration file for the anti-virus engine.

Anti-Spam-Config Publication

The Anti-Spam-Config publication contains the configuration files that determine anti-spam functionality.

- `offensive-words` (list)
- `trusted-relays` (list)
- `spam.conf` (configuration file)
- `spam.d/net.conf` (configuration file)
- `spam.d/dnsbl.conf` (configuration file)
- `spam.d/spamassassin.conf` (configuration file)
- `spam.d/re.rules` (custom rules)
- `spam.d/db.force` (alterations to rule weights and enabled state)
- `spam.d/db.pdelta` (rule probabilities)
- `spam.d/compile.d/compiler.conf` (general rule compiler configuration)
- `spam.d/compile.d/destination.conf` (Known Spam Destinations feature group configuration)
- `spam.d/compile.d/heuristic.conf` (Heuristic Analysis feature group configuration)
- `spam.d/compile.d/sender.conf` (Sender Reputation feature group configuration)
- `spam.d/compile.d/site.conf` (Site Features feature group configuration)

Policy Publication

The Policy publication contains the policy script and the most commonly used lists and maps.

- `policy.siv` (policy script)
- `internal-hosts` (list)
- `whitelisted-hosts` (list)
- `whitelisted-senders` (list)
- `blacklisted-hosts` (list)
- `blacklisted-senders` (list)

DomainKeys-Identified-Mail Publication

The DomainKeys-Identified-Mail publication contains the `dkim.conf` configuration file, which specifies the required signing options, and the location of the private key that is used to create the DKIM signature.

- `/opt/pmx6/etc/dkim.conf` (configuration file)

Related concepts

[Managing Publications](#) (page 184)

Related information

[pmx-share](#)

3.4 Administrative Groups

This section describes the setup and Management of the Groups Web Interface, which a system administrator can use to delegate selected tasks to other system administrators.

The groups management feature makes it possible to create a customized web-based interface that permits system administrators to access a subset of PureMessage functionality. These sub-administrators (*group administrators*) can be given access rights by *global administrators* to manage select features of PureMessage, such as the quarantine, reports, lists and limited aspects of the PureMessage policy. Group administrators are often responsible for specific domains or groups of recipients within an organization.

Using a series of command-line tools, the global administrator can:

- create a group
- define the recipient addresses/domains that make up the group
- create administrator accounts
- grant administrators access rights to manage one or more groups
- define group access rights

The ability to create groups gives an organization more flexibility in the way it deals with email filtering. For example:

- An administrator can be given access rights to multiple domains within the organization, with a different set of permissions for each domain.
- A "helpdesk" group can be defined that allows access to all relevant data, but does not permit account-holders to change policy options or edit lists.

Access rights are assigned on the basis of group/administrator pairs. First, the global administrator creates groups and group administrator accounts. Then the global administrator creates an association between the groups and the administrator accounts so that an administrator has access to manage one or more groups. See the Groups Setup tutorial for more about the steps required and the order in which they are performed. The group administrators GUI will reflect the access rights granted by the global administrator. Once it is configured, group administrators can use a supported browser to access the Groups Web Interface at `https://<Hostname>:28443/groups`. Global administrators can do the same by way of a full-access administrator account. For more information, see "Creating a Full-Access Administrator Account".

Important

Create and store all of the data related to groups, administrator accounts and access rights on the central server (CSM) so that it can be properly distributed to edge servers.

The global administrator uses the following command-line programs to set up and configure groups:

- `/opt/pmx6/bin/pmx-group` : Used for adding and deleting groups, assigning administrators to groups, listing group details, creating group permissions, and modifying group permissions for PureMessage services running on remote servers.
- `/opt/pmx6/bin/pmx-group-file` : A group file management utility used to add and delete group-specific documents and banners.
- `/opt/pmx6/bin/pmx-group-list` : A group list management utility used to add and delete group lists.

- `/opt/pmx6/bin/pmx-group-policy` : A utility for adding and deleting group-specific policy settings.
- `/opt/pmx6/bin/pmx-user` : Used to create and delete administrator accounts, this command also has an option to create a full-access administrator account that has access rights to all configured groups, as well as data that is not group-specific.

Related tasks

[Creating a Full-Access Administrator Account](#) (page 207)

[Generating a Self-Signed Certificate for the Groups Web Interface](#) (page 229)

Related information

[pmx-group](#)

[pmx-group-file](#)

[pmx-group-list](#)

[pmx-group-policy](#)

[pmx-user](#)

3.4.1 How Groups Work

Groups, group administrators, and permissions are all defined in the database, but members of a group are defined in a group's member list, which is stored in a subdirectory of the `/opt/pmx6/members-per-group` directory. For example, member lists Group1 and Group2 could appear as follows:

```
/opt/pmx6/etc/members-per-group/g/r/group1
/opt/pmx6/etc/members-per-group/o/t/othergroup
```

Notice that group lists are located within automatically generated subfolders whose names are based on the first two letters of the group name. This is to limit the number of files in a single folder in very large systems.

The elements of the list are email address parts, which take one of the following three forms:

- Exact: `ie/ user@sub.domain.com`
- User Inexact: `ie/ user@`
- Domain Inexact: `- ie/@sub.domain.com`

The `members-per-group` directory is a default resource that is currently populated manually by the global administrator. This can be done by running the `pmx-list` command, editing a list directly, or using `pmx-ldap-sync` or some other tool to populate the list from an external source. The global administrator must then manually synchronize the changes to the `members-per-group` directory with the database via the `pmx-profile sync-to-db` and `pmx-makemap --grouplist -g` commands.

Note

PureMessage uses `pmx-profile` and `pmx-makemap --grouplist -g` to regularly synchronize data with edge servers. If these jobs are enabled via the **Local Services** tab of the PureMessage Manager, it is not necessary to run these commands manually.

Recipient-to-Group Mappings

The `pmx-makemap --grouplist -g` command converts the contents of the `members-per-group` directory into one flat CDB map that maps recipients to groups. For example, it might look like this:

- `joe@domain.com: group1`
- `@domain.com: group2`
- `postmaster@: group3`
- `frank@sub.domain.com: group4`

Every recipient entry in the `members-per-group` directory has a corresponding entry in the CDB map, with the right-hand column indicating the short name of the group that it is a member of. The CDB map defines the recipient-to-group mappings used by the PureMessage policy.

When the policy processes a message, the mappings are applied and data concerning the recipient/group relationships are written the `message_log` and the quarantine minfo log. This data cannot be retroactively assigned or changed after the message has been processed by the policy.

Two things are important in properly determining the recipient-to-group mappings for a message: traffic direction and precedence.

Traffic Direction

Mappings are evaluated only for the sender (ENVELOPE FROM address) if a message is outbound, and for all recipients (ENVELOPE TO addresses) if the message is inbound. Traffic direction is determined based on the "internal-hosts" list, so if a message comes from a relay that is in the internal hosts list, it is expected that the message is outbound. Otherwise it is inbound.

Note

If the ENVELOPE TO address is mapped to another address in the recipient-aliases map, the mapped address is used for group determination, rather than the original ENVELOPE_TO address.

Precedence

A recipient-to-group mapping must be unique, so precedence is enforced in the determination of a group. The lookup works according to the following matching order:

1. Exact: (for example, `joe@domain.com`)
2. User Inexact: (for example, `user@`)
3. Subdomain Inexact: (for example, `@sub.domain.com`) The example below shows the precedence used for subdomains.
4. Domain Inexact: (for example, `@domain.com`)

This means that if a message is to `joe@a.b.c.domain.com` for example, the following matches would be tested against, with the first one returning:

- `joe@a.b.c.domain.com`
- `joe@`
- `@a.b.c.domain.com`
- `@c.domain.com`
- `@domain.com`
- NO MATCH

The subsequent group information is then written to the `message_log` and the quarantine minfo (if any mapping exists for the addresses it checked). This information is then consumed by other scheduled jobs in the system (for example, `pmx-qmeta-index` and `pmx-reports-consume-message-log`) and used, for example, for group-specific reports and the quarantine.

Creating a Group

The global administrator creates a group to store the email addresses or domains of recipients whose mail will be managed by the group administrator responsible for that group. Groups must be created on the central server (CSM). Group names must:

- contain only alphanumeric characters, hyphens, and underscores (all other characters are invalid)
- not begin with a number, hyphen, or underscore
- consist of at least two characters

To create a group:

1. Log on to the central server as the “pmx6” user.
2. Run the following command:

```
pmx-group --add --group <Name> --description <GroupDescription>
```

where *Name* is the name of the group and *GroupDescription* is additional descriptive text about the group. Be sure to enclose the *Name* and *GroupDescription* text strings in quotation marks if the strings contain spaces.

The newly created group is stored in a subdirectory of the `/opt/pmx6/etc/members-per-group` directory. When you add a group, single-letter subdirectories are automatically created based on the first two letters of the group name. This is done to ensure that the maximum files per directory limit is not exceeded in cases where organizations have large numbers of groups.

3. Run `pmx-makemap --grouplist -g` to create a CDB list that can be synchronized with the database.
4. Run `pmx-profile sync-to-db` to synchronize the list data with the database.

Note

PureMessage uses `pmx-profile` and `pmx-makemap --grouplist -g` to synchronize data with edge servers. If these jobs are enabled via the **Local Services** tab of the PureMessage Manager, it is not necessary to run these commands manually.

See the `pmx-group` man page (`man pmx-group`) for more information.

Deleting a Group

To delete a group:

1. Log on to the central server as the “pmx6” user.
2. Run the following command:

```
pmx-group --delete --group <Name>
```

where *Name* is the name of the group as it appears in the `/opt/pmx6/etc/members-per-group` directory.

3. Run `pmx-makemap --grouplist -g` to create a CDB list that can be synchronized with the database.
4. Run `pmx-profile sync-to-db` to synchronize the list data with the database.

Note

PureMessage uses `pmx-profile` and `pmx-makemap --grouplist -g` to synchronize data with edge servers. If these jobs are enabled via the **Local Services** tab of the PureMessage Manager, it is not necessary to run these commands manually.

See the `pmx-group` man page (`man pmx-group`) for more information.

Adding Members to a Group

The email recipients associated with a particular group must have membership in that group. Members are included in a group by adding their email addresses. This can be done by either editing the group file directly or using the `pmx-list` command. The groups are stored in the `members-per-group` directory.

Members must be specified as email address parts; that is, they must take the form of individual addresses, subdomains, domains, or the first part of the email address. For example:

- *JaneD@example.com*
- *@subdomain.example.com*
- *@example.com*
- *Jane@*

Important

Any given recipient should be added to one group only. Adding a recipient address to more than one group may produce unpredictable results when the PureMessage policy processes the address.

- To add members by editing the group file:
 - a) At the command line, navigate to the `/opt/pmx6/etc/members-per-group` directory.
 - b) Using an editor (for example, `vi`), enter the email addresses of those who will administer the group.
 - c) Run `pmx-makemap --grouplist -g` to create a CDB list that can be synchronized with the database.
 - d) Run `pmx-profile sync-to-db` to synchronize the list data with the database.
- To add members using the `pmx-list` command:
 - a) At the command line, as the “pmx6” user, run the following command:

```
pmx-profile sync-from-db --resource=members-per-group
```

- b) Run the following command:

```
pmx-list add members-per-  
group#<GroupName> address1@example.com address2@example.com ...
```

where *GroupName* is the name of the group to which addresses are added. The addresses are added to the group file in a subdirectory of the `/opt/pmx6/etc/members-per-group` directory.

- c) Run `pmx-profile sync-to-db`.
- d) Run `pmx-makemap --grouplist`.

Note

PureMessage uses `pmx-profile` and `pmx-makemap --grouplist -g` to synchronize data with edge servers. If these jobs are enabled via the **Local Services** tab of the PureMessage Manager, it is not necessary to run these commands manually.

See the `pmx-list` man page for more information.

Deleting Members from a Group

To delete members from a group:

1. At the command line, as the “pmx6” user, run the following:

```
pmx-list remove members-per-group#<GroupName>
address1@example.com address2@example.com ...
```

The addresses are deleted from the group file that is located in a subdirectory of the `/opt/pmx6/etc/members-per-group` directory.

2. Run `pmx-makemap --grouplist -g` to create a CDB list that can be synchronized with the database.
3. Run `pmx-profile sync-to-db` to synchronize the list data with the database.

Note

PureMessage uses `pmx-profile` and `pmx-makemap --grouplist -g` to synchronize data with edge servers. If these jobs are enabled via the **Local Services** tab of the PureMessage Manager, it is not necessary to run these commands manually.

See the `pmx-list` man page (`man pmx-list`) for more information.

Creating an Administrator Account

Permissions are assigned on the basis of group/administrator pairs. It is therefore necessary to create the accounts the administrators will use to access the groups.

To create an administrator account:

1. At the command line, as the “pmx6” user, run the following:

```
pmx-user --add --username <Username> --fullname <Fullname> --email
<Address>
```

where *Username* is the system name that you assign, *Fullname* is the user’s actual name and *Address* is the email address that has been specified in a group.

You are prompted to enter a password.

2. Type in a password for this administrator, and press **Enter** .

You are prompted to verify the password.

3. Re-enter the password.

A message is displayed advising that the user has been added.

See the `pmx-user` man page for more information.

Creating a Full-Access Administrator Account

If, as a global administrator, you want to have access to all enabled features of the Groups Web Interface. You can create a special “superuser” account that grants you access rights for all configured groups and their associated permissions.

Note

A full-access administrator account is automatically created during PureMessage installation. The account requires the same username and password set for the PureMessage Manager during installation. Follow the procedure below if you want create additional full-access accounts.

Another benefit of a superuser account is that it becomes possible to access all data that is not specific to groups. For example, an administrator may want to take advantage of the reporting and quarantine search features available in the Groups Web Interface as an alternative to the equivalent features in the PureMessage Manager. As with other administrator accounts, access rights can be disabled and enabled. So, for example, if access to the **Configuration** tab is not necessary, you can disable it. When setting access rights for a superuser account, however, you *do not* specify a group.

To create a full-access account:

1. At the command line, as the “pmx6” user, run the following:

```
pmx-user --add --username <UserName> --fullname <User> --email
<Address> --superuser
```

where `<UserName>` is a name you assign, `<User>` is the user's actual name, and `<Address>` is the email address associated with this account.

You are prompted to enter a password.

2. Enter a password and confirm it.

A message is displayed advising that the user has been added.

See the `pmx-user` and `pmx-group` man pages for more information.

Deleting an Administrator Account

To delete an administrator account:

At the command line, as the 'pmx6' user, run the following:

```
pmx-user --delete --username <Username>
```

where `Username` is the username originally assigned to the administrator.

A message is displayed indicating that the account has been removed.

See the `pmx-user` man page (`man pmx-user`) for more information.

Adding an Administrator Account to a Group

Access rights are granted on the basis of group/administrator pairs. Before setting access rights for a group, you must associate the group with one or more administrator accounts. A group can be managed by more than one administrator, and a single administrator can manage multiple groups.

To add an administrator account to a group:

At the command line, as the “pmx6” user, run the following:

```
pmx-group --add-admin --group <GroupName> --user <Username>
```

where *GroupName* and *Username* are the previously assigned names that you now want to associate.

A message is displayed advising that the specified user can now administer this group.

See the `pmx-group` man page (`man pmx-group`) for more information.

Removing an Administrator Account from a Group

Access rights are granted on the basis of group/administrator pairs. You can use the `pmx-group` command to remove an administrator account from a group, severing a previously created group/administrator pair.

To delete an administrator account from a group:

At the command line, as the “pmx6” user, run the following:

```
pmx-group --delete-admin --group <GroupName> --user <Username>
```

where *GroupName* and *Username* represent the previously associated group and administrator account.

A message is displayed advising that the specified user can no longer administer this group.

See the `pmx-group` man page (`man pmx-group`) for more information.

Setting Group Access Rights

Access rights are set on the basis of group/administrator pairs. By default, any group that has been associated with an administrator account has full access rights enabled. Use the `pmx-group` command to enable or disable permissions. These permissions correspond with tabs and options in the Groups Web Interface. Most permissions are enabled/disabled by specifying "on" or "off" with the `--value` option of the command described below. The exceptions are the list options, which have "off", "read" and "write" as potential values. If you want to grant both read and write permissions to a group administrator, set `--value` to "write"

You can enable/disable specific options or entire sets of options (tabs). For example, if you do not want an administrator to have access to any reports, you can exclude the **Reports** tab from this administrator's GUI. To do this, specify "reports" for `--permission` and "off" for `--value` in the command described below. To disable an individual report, you could run a similar command that refers to a specific report (for example, "reports.topreleasers").

Note

When setting access rights for a “superuser” account, you *do not* specify a group.

To set group access rights:

At the command line, as the “pmx6” user, run the following:

```
pmx-group --set-perm --group <GroupName> --user <Username> --permission
<PermissionName> --value <ValueName>
```

where *GroupName* and *Username* are the previously assigned names; *PermissionName* is the full permission name (as shown when `pmx-group` is run with the `--view-perm` option); and *ValueName* is the permission setting (usually “on” or “off”).

The permission is enabled or disabled for the specified user.

See the `pmx-group` and `pmx-user` man pages, and [Configuring Full Access to the Group Administrator Interface](#) for more information.

Related concepts

[Viewing Group Access Rights](#) (page 209)

Viewing Group Access Rights

When a group is created and associated with at least one administrator account, full access rights are enabled by default. These rights can be modified and viewed using the `pmx-group` command. You can view all of the access rights for a group/administrator pair, or view a specific permission for a group/administrator pair.

Related tasks

[Setting Group Access Rights](#) (page 208)

Viewing All Access Rights

To view complete access rights for a group/administrator pair:

At the command line, as the “pmx6” user, run the following:

```
pmx-group --view-perm --group <GroupName> --user <Username>
```

where *GroupName* and *Username* are the names you assigned previously.

A complete list of access rights is displayed, with the status of each permission shown to the right of the permission name.

Viewing a Specific Access Right

To view the status of a specific permission for a group/administrator pair:

At the command line, as the “pmx6” user, run the following:

```
pmx-group --view-perm --group <GroupName> --user <Username> --permission
<PermissionName>
```

where *GroupName* and *Username* are the previously assigned names, and *PermissionName* is the full permission name (as shown when `pmx-group` is run with the `--view-perm` option).

The status of the specified permission is displayed.

Creating a Policy Setting

By default, the **Policy: Policy Settings** page of the **Configuration** tab has no settings configured. You can create a custom policy setting, in the form of a check box, that gives group administrators the

ability to enable/disable group-specific policy filtering. Policy settings are created from the command line using `pmx-group-policy`.

To create a policy setting:

1. At the command line, as the “pmx6” user, run the following:

```
pmx-group-policy --add --id <ID> --name <ListName>\
--type <UI_Control> --style <ListType>\
--description <ControlName>
```

where *ID* is a unique identifier for the list, *ListName* is the name that appears in the policy constructor in the PureMessage Manager, *UI_Control* is the type of control the group administrator will use to enable/disable the setting (only “checkbox” is currently supported), *ListType* specifies whether the related list is an “optin” or “optout” list, and *ControlName* is the text label for the control that appears in the Groups Web Interface.

2. Run `pmx-profile sync-from-db`.

A check box option is now displayed on the **Policy: Policy Settings** page of the **Configuration** tab. In addition, a new group list is added to `/opt/pmx6/etc/lists.conf` and is also available in the drop-down list that is used for specifying lists in the PureMessage Manager's policy constructor.

For additional information, see the `pmx-group-policy` man page and “Adding and Defining a Policy Setting” in the Groups Setup tutorial.

For more information about configuring the PureMessage policy, see “Policy Configuration” in the *Administrator's Reference* and “Policy” in the *Manager Reference*.

Related tasks

[Tutorial: Adding and Defining a Policy Setting](#) (page 235)

Creating a Group List

By default, the **Configuration** tab of the Groups Web Interface contains several lists that can be used to determine how the policy deals with specified email addresses and/or domains. These lists are **Allowed Relays**, **Allowed Senders**, **Blocked Relays** and **Blocked Senders**. You can also create custom lists to augment or replace the existing lists using the `pmx-group-list` command.

Regardless of whether the list is a default list or a custom list, it must be combined with a rule in the PureMessage policy in order to have any effect. For more information, see “Creating a Custom Group Policy List and Rule” in the Groups Setup tutorial.

To create a group list:

1. At the command line, as the “pmx6” user, run the following:

```
pmx-group-list --add --id <ID> --name <Name> --description
<Description> --match-type <MatchType>
```

where *ID* is a unique identifier for the list, *Name* is the list name, *Description* is additional details about the list and *MatchType* is the valid list match type as specified in the policy (types include “is”, “contain”, “match”, “matches”, “re”, “domain”, “mail”)

2. Run `pmx-profile sync-from-db`.

A list with the assigned name is added to `/opt/pmx6/etc` and list data is added to `etc/multilists.conf`.

- For each server that is running the Groups Web Interface, at the command line, run:

```
pmx-httpd restart
```

The list will be accessible the next time the group administrator views the **Configuration** tab.

See the `pmx-group-list` man page (`man pmx-group-list`) for more information.

Related tasks

[Deleting a Group List](#) (page 211)

[Creating a Custom Group Policy List and Rule](#) (page 233)

Deleting a Group List

The **Configuration** tab of the group administrator GUI contains several default lists, along with any custom lists that you may have added. You can remove lists using the `pmx-group-list` command.

To delete a group list:

- At the command line, as the “pmx6” user, run the following:

```
pmx-group-list --delete --id <ID>
```

where *ID* is a unique identifier for the list.

A list will be removed from `etc/multilists.conf` the next time PureMessage resources are synchronized from the database via the resource-sync scheduled job.

- For each server that is running the End User Web Interface, at the command line, run:

```
pmx-httpd restart
```

The list will be accessible the next time a group administrator views the **Configuration** tab.

See the `pmx-group-list` man page (`man pmx-group-list`) for more information.

Related tasks

[Creating a Group List](#) (page 210)

Adding and Deleting Custom Quarantine Search Reasons

In the Groups Web Interface, the **Search Parameters** sidebar of the **Search** tab has a drop-down list from which you can select a **Quarantine** search. One of the quarantine search options allows you to select a reason that the message was quarantined from the **Reason** drop-down list.

Global administrators can also add custom reasons to the default set of reasons using the `pmx-group` command. For example, a global administrator may want to add the reason “Offensive” so that group administrators have the option of searching for messages that were quarantined because they contain offensive language. Once enabled, a custom reason access right can be turned on or off in the same manner as other access rights.

- To add a custom reason:

At the command line, as the “pmx6” user, run the following:

```
pmx-group --add-perm --permission quarantine.reason.<ReasonName>
```

A message is displayed, advising that the permission has been added. The new reason will appear in the **Reason** drop-down list for all group administrators who have been granted access to this feature.

- To delete a custom reason:

At the command line, as the “pmx6” user, run the following:

```
pmx-group --delete-perm --permission quarantine.reason.<ReasonName>
```

A message is displayed, advising that the permission has been deleted. The reason will no longer appear in the **Reason** drop-down list.

Note

The `pmx-group` command only permits the addition and removal of `quarantine.reason` permissions.

See the `pmx-group` man page (`man pmx-group`) for more information.

Related concepts

[Viewing Group Access Rights](#) (page 209)

Related tasks

[Setting Group Access Rights](#) (page 208)

Adding and Deleting Custom Log Search Reasons

In the Groups Web Interface, the **Search Parameters** sidebar of the **Search** tab has a drop-down list from which you can select a **Logs** search. One of the log search options is the **Reason** drop-down list, which allows you to select the reason that the message was quarantined.

Global administrators can add to the default set of reasons by editing the PureMessage policy and adding new entries to the **Log Reasons** policy list. For example, a global administrator may want to add the reason “Confidential” so that group administrators have the option of searching for messages that were quarantined because they contain confidential information.

- To edit the policy:

You must create a policy rule that quarantines messages with the specified custom reason (in this case, “Confidential”).

- Using the Policy Constructor (default page of the **Policy** tab), create the test portion of the rule.
- From the **Execute actions and rules** drop-down list, select **Log the message with key/value pair**.
- In the first adjacent field, type `pmx_reason`.
- In the second adjacent field, type `Confidential`.
- Click **Save**.
- Click **Commit**.

Alternatively, edit the rule by adding the following line directly to the `policy.siv` script:

```
pmx_mark "pmx_reason" "Confidential"
```

- To add a reason to the drop-down list:

You must add the new reason (in this case, “Confidential”) to the list of existing search reasons.

- a) In the **Lists** section of the **Policy** tab sidebar, click **Log Reasons**.
- b) In the **Add Items** text box, type the custom reason, and click **Add**.

The reason is displayed alphabetically among the **List Items** and will be displayed in the **Reason** drop-down list in Groups Web Interface when the log search option is selected.

- To delete a reason from the drop-down list:
 - a) In the **Lists** section of the **Policy** tab sidebar, click **Log Reasons**.
 - b) From the **List Items**, select the check box next to the reason that you want to remove, and click **Delete**.

The reason is removed from the list.

Related concepts

[Editing the Policy](#) (page 80)

[Viewing Group Access Rights](#) (page 209)

Related tasks

[Editing Lists](#) (page 120)

[Setting Group Access Rights](#) (page 208)

Creating a Group-Specific Template

You can create group-specific templates with the `pmx-group-file` command that can be made available to group administrators via the **Configuration** tab of the Groups Web Interface.

The global administrator can use this feature to create options for group-specific banners (see [tutorial example](#)) and group-specific documents. Links to these templates are displayed on the **Configuration** sidebar; group-specific banners appear under a Banners heading, while all other templates appear under a Documents heading.

By default, both the document and banner templates are displayed in the group administrator interface as a blank text box, along with a **Save** button. If you, as the global administrator, want to add read-only content to a document or banner template, you must log into the group administrator interface to enter and save the content. You must then set the access rights as necessary to make the template available to the appropriate group(s). See the "Related tasks" section below for more information.

To create a group-specific template:

At the command line, as the "pmx6" user, run the following:

```
pmx-group-file --add --id <ID> --name <TemplateName>\
--type <TemplateType> --charset <Encoding>
```

where `<ID>` is the unique identifier for the template, `<TemplateName>` is the name that appears on the **Configuration** sidebar, `<TemplateType>` specifies whether it is a banner or document, and `<Encoding>` is the encoding type the banner uses (for example, Latin1). If not specified, default type is "document" and the default charset is "latin1".

A message is displayed advising that the template has been added. The new template will now be visible on the **Configuration** sidebar.

For additional information, see the `pmx-group-file` man page.

Related tasks

[Creating a Full-Access Administrator Account](#) (page 207)

[Setting Group Access Rights](#) (page 208)

[Tutorial: Adding a Group Banner](#) (page 238)

Deleting a Group-Specific Template

You can delete group-specific templates with the `pmx-group-file` command.

The templates are displayed on the **Configuration** sidebar in the form of group-specific banners and group-specific documents.

To delete a group-specific template:

1. At the command line, as the “pmx6” user, run the following:

```
pmx-group-file --del --id <ID>
```

where `<ID>` is the unique identifier that was assigned when the template was created.

A message is displayed advising that the template has been deleted.

2. Run `pmx-httpd restart`.

The new template will no longer be visible on the **Configuration** sidebar.

For additional information, see the `pmx-group-file` man page.

Related tasks

[Creating a Group-Specific Template](#) (page 213)

3.4.2 Viewing and Managing Search Results

You can view the results for quarantine and log searches in the **Search Results** pane. The quarantine **Search Results** pane has additional options for approving, forwarding, saving, and deleting messages.

Note

The search options described in this section are only available if the Groups Web Interface access rights for these options are enabled. All reports and options are enabled by default. To access the Groups Web Interface, point a supported browser at `https://<Hostname>:28443/groups`. See "Creating a Full-Access Administrator Account" and the Groups Setup tutorial for more information.

Related concepts

[Administrative Groups](#) (page 201)

This section describes the setup and Management of the Groups Web Interface, which a system administrator can use to delegate selected tasks to other system administrators.

[Tutorial: Groups Setup](#) (page 229)

Related tasks

[Creating a Full-Access Administrator Account](#) (page 207)

Quarantine Search

Quarantine is the default search type displayed on the **Search Parameters** sidebar. The quarantine contains messages whose delivery has been suspended, typically because they were identified as spam or they have violated content rules. After searching the quarantine, you can examine the results, and release, forward, save, or delete messages.

Searching the Quarantine

Search the quarantine by building queries with the options available on the **Search Parameters** sidebar. Results are displayed in the **Search Results** pane, where they can be viewed, approved, forwarded, saved, deleted, or reported to [SophosLabs](#) as *false positives*.

To search the quarantine:

1. Define the parameters for your quarantined message search by setting one or more of the following:

Note

The text boxes support string-based searches. All text boxes except **ID** support the “*” and “%” wildcards. When using wildcards to find all or part of a domain, all search terms must be preceded by the “@” symbol. Searches are case-insensitive.

- **Sender:** Enter a full or partial sender’s email address.
- **Recipient:** Enter a full or partial recipient’s email address.
- **Subject:** Enter the full subject or keywords contained in an email’s subject line.
- **Start Date Range:** Type the date and time in YYYY-MM-DD [hh:mm] format. If you do not specify hours and minutes, the start time is 0:00.
- **End Date Range:** Type the date and time in YYYY-MM-DD [hh:mm] format. If you do not specify hours and minutes, the end time is 23:59.
- **Relay:** Enter a full or partial hostname or IP address of an internal mail relay.
- **ID:** Enter an identification number assigned to a message during processing. You can use this text box to search for the following types of message ID numbers:
 - **Queue ID:** PureMessage assigns a unique Queue ID to each message in the quarantine. To view a message’s Queue ID, click the message subject, and then click **Info**.
 - **Quarantine Digest ID:** When PureMessage generates quarantine digests, each message listed in the digest is assigned an ID code.
 - **Header Message ID:** If the format of the search string resembles a header message ID (for example, by containing an “@” separator), the contents of this field are tested against the value in the Message-ID header. To view a message’s Message-ID header, click the message subject, and then click **Source**.
- **Reason:** From the drop-down list, select the reason that the email was quarantined. The available options include Spam, Blacklisted, Virus, and Suspect. The default is to search for any of these reasons.
- **Results to Display:** From the drop-down list, select the number of results rows to display at one time. The minimum is 20 and the maximum is 1,000.

2. Click **Search**.

Matches for the search are displayed in a table in the **Search Results** pane. The total number of results is displayed at the bottom left of the pane.

Viewing Quarantine Search Results

The following general options are available for viewing search results:

- Click the “up/down” arrow next to a search results column heading to order the displayed results alphanumerically in that column. Click the “up/down” arrow button again to toggle the results between ascending and descending order.
- If multiple pages of search results are available, use the controls at the top right of the content pane to view the additional pages.
- Click the hyperlinked text in a message row to view further details of that message.

Viewing Message Details

You can view additional information about a specific message in the **Message Details** dialog box.

To view message details:

1. In the **Search Results** pane, click the message subject.
2. Click one of the following buttons:
 - **Content**: Displays the body of the message.
 - **Source**: Displays the raw source of the message.
 - **Attach**: Displays the name, file extension, file type and size of any message attachments. For more information, see “Downloading Attachments”.
 - **Info**: Displays summary information for the message, including the quarantine ID, the message size, the message’s spam probability, envelope from and to values, and the message’s queue id.
 - **Status**: Indicates what actions have been performed on the message. For more information, see “Viewing Status Details”.

The selected information is displayed in the **Message Details** dialog box.

Downloading Attachments

You can download message attachments via the **Attach** tab of the **Message Details** dialog box.

CAUTION

Attachments may have malicious or viral content. Always take appropriate precautions when downloading attachments, especially those associated with messages identified as viruses, suspicious attachments and spam.

To download attachment details:

1. In the **Search Results** pane, click the message subject.

The **Message Details** dialog box is displayed.

2. Click the **Attach** tab.
3. Click the attachment name.

The download management dialog box for your default browser is displayed.

4. Use your web browser’s “download file” feature to specify the download location.

Viewing Status Details

The icon in the **Status** column of the **Search Results** pane represents a summary of the actions, if any, that have been performed on a message. The **Status** page of the **Message Details** dialog box shows, in detail, the actions that have been taken for each recipient of the message. For example, the **Status** page may indicate that one recipient has deleted the message, a second recipient has approved the message, and a third recipient has taken no action. The status is a record of actions taken by administrators and end users.

For each recipient listed on the **Status** page, there is an **Approve** button that can be used to “force-approve” the message for an individual recipient. For example, this lets you re-release a message to a recipient for which the message has already been approved. This is different from approving a message on the **Search Results** pane, where messages can only be approved for recipients for which the message has not already been approved or deleted.

You can also save, forward or delete a message or multiple messages from within the **Message Details** dialog box.

A message is displayed in the **Search Results** pane with one of the following status icons:

	The message has been approved for all recipients.
	The message has been approved for some recipients.
	The message has been deleted for all recipients.
	The message has been deleted for some recipients.
	The message has been approved or deleted for all recipients.
	The message has been approved or deleted for some recipients but no action has been taken for other recipients.
	No action has been taken for any of the recipients.

- To view status details for a specific message:
 - Click the status icon for that message.
The **Status** page is displayed, which shows the address, group and status for each recipient.
- To approve a message:
 - Click the **Approve** button next the recipient that you want to approve.
The message is approved for that recipient regardless of the current status.
- To save, forward or delete a message:
 - Select one or more check boxes, and click the appropriate button at the bottom of the page to approve, forward, save or delete the message(s).

Managing Quarantine Search Results

A row of buttons at the bottom of the **Search Results** pane allows you to perform a variety of actions on a selected message or group of messages. You can report misclassified messages to Sophos, approve messages for delivery, forward messages to one or more recipients, save local copies of messages, and delete messages.

Approving a Message

Approving messages releases them for delivery to the intended recipients. A copy is sent to the approved recipient, and another copy is held in the quarantine. Quarantined messages are eventually deleted (or archived) by a scheduled service.

Note

If this message is addressed to multiple recipients, and the status of actions performed differs for the various recipients, the message will only be approved for recipients for which the message has yet to be approved or deleted. To “force-approve” a message for a recipient, you must click the **Approve** button next the recipient’s address on the **Status** page of the **Message Details** dialog box.

To approve a message:

- Select the check box for the message that you want to approve.
- Click **Approve**.

The message is released to its intended recipient(s).

Forwarding a Message

When you forward a message, a copy is sent to one or more specified recipients, and another copy remains in the quarantine. Clicking **Forward** beneath the quarantine **Search Results** pane launches a dialog box with options for reporting mis-classified messages to Sophos or specifying recipient email addresses.

To forward a message:

1. Select the check box for the message(s) that you want to forward.
2. Click **Forward**.

The **Forward** dialog box is displayed, prompting you to either report the message(s) to Sophos Labs or specify recipient email addresses.

3. In the **Forward** dialog box, choose one of the following forward options:
 - Select the appropriate option button to report a message as a false positive, false negative or potential virus.
 - Type the email addresses of recipients (one per line), and, optionally, type a brief explanation in the **Comments** text box.
4. Click **Forward**.

Saving a Message

The “Save” feature allows you to save a copy of the message on your local machine in mbox format using your default web browser.

To save a message:

1. Select the check box for the message(s) that you want to save.
2. Click **Save**.
3. Use your web browser’s file download feature to specify where the file is saved.

Deleting Messages

The **Delete** and **Delete All** commands permanently remove messages from the quarantine. Deleted messages are eventually deleted (or archived) via a scheduled job.

To delete a message:

1. Select the check box for the message that you want to delete.
2. Click **Delete**.

To delete all messages returned by the current query, click **Delete All**.

Log Search

Select **Logs** from the drop-down list at the top of the **Search Parameters** sidebar to view search options. The log search allows you to search for records of past messages. The mail and message logs maintain data about how the mail transfer agent (MTA) and PureMessage have dealt with specific messages, providing a means for evaluating the effectiveness of the current mail-filtering policy. Messages listed in the logs can be searched and examined. For example, you might search and analyze the logs for the following reasons:

- You want to confirm that the policy options you have set are working as expected.
- A user has reported that a message has not been delivered and wants to know why.

Note

Log searches are only available if you selected the Postfix mail transfer agent (MTA) during PureMessage installation. PureMessage uses the Log Search Index background service to index log data for faster searching. If you opted for the sendmail MTA during installation or are a Postfix user who upgraded to the latest PureMessage from a previous version of PureMessage, the Log Search Index service is disabled by default. For more information about this service, see the `pmx-logsearch-index` and `logsearch.conf` man pages.

Related information

[pmx-logsearch-index](#)

[logsearch.conf](#)

Searching the Logs

Search the logs by building queries with the options available on the **Search Parameters** sidebar. Results are displayed in the **Search Results** pane, where they can be viewed.

To search the logs:

1. Define the parameters for your logs search by setting one or more of the following:

Note

The text boxes support string-based searches. All text boxes except **ID** support the "*" wildcard. Searches are case-insensitive

- **Sender:** Enter a full or partial sender's email address.
- **Recipient:** Enter a full or partial recipient's email address.
- **Start Date Range:** Type the date and time in YYYY-MM-DD [hh:mm] format. If you do not specify hours and minutes, the start time is 0:00.
- **End Date Range:** Type the date and time in YYYY-MM-DD [hh:mm] format. If you do not specify hours and minutes, the end time is 23:59.
- **Host:** Enter a full or partial hostname or IP address of a host involved in the transmission of a message.
- **ID:** Enter an identification number assigned to a message during processing. You can use this text box to search for the following types of message ID numbers:
 - **Queue ID:** PureMessage assigns a unique Queue ID to each message in the quarantine. To view a message's Queue ID, click the message subject, and then click **Info**.
 - **Quarantine Digest ID:** When PureMessage generates quarantine digests, each message listed in the digest is assigned an ID code.
 - **Header Message ID:** If the format of the search string resembles a header message ID (for example, by containing an "@" separator), the contents of this field are tested against the value in the Message-ID header. To view a message's Message-ID header, click the message subject, and then click **Source**.
- **Action:** From the drop-down list, select the action that caused the message to be logged. The available options include **Deliver**, **Discard**, **Quarantine** and **Reject**. The default is to search for any of these actions.
- **Reason:** From the drop-down list, select the reason that the email was logged. The available options include Allow List, Block List, Blocked, Offensive, Spam and Virus. The default is to search for any of these reasons.

- **Results to Return:** From the drop-down list, select the number of results rows to display at one time. The maximum is 1,000.
2. Click **Search**.

Matches for the search are displayed in a table in the **Search Results** pane. The total number of results is displayed at the bottom left of the pane.

Viewing Log Search Results

Search results are displayed in a series of columns in the Content pane. For each message, the **Date/Time**, **Sender**, **Recipient**, **Subject**, **Action** and **Reason** are shown. If the message is addressed to multiple recipients, resulting in PureMessage performing different actions for each recipient, the **Action** and/or **Reason** displayed is "Multi". Processing details for the individual recipients are shown in the **Activity** section of the **Info** tab in the **Message Details** dialog box.

The following general options are available for viewing search results:

- Click the "up/down" arrow next to a search results column heading to order the displayed results alphanumerically in that column. Click the "up/down" arrow button again to toggle the results between ascending and descending order.
- If multiple pages of search results are available, use the controls at the top right of the content pane to view the additional pages.
- Click the hyperlinked **Subject** text in any log entry row to view further details of that entry.

Viewing Message Details

You can view additional information about a specific log entry in the **Message Details** dialog box. This dialog box contains two tabs: **Info** and **Raw logs**. The **Info** tab is displayed by default.

The following basic details are shown on the **Info** tab:

- **Mapped:** The email address the message was sent to, along with any alias addresses. If the recipient address has no aliases, then it is the only address shown.
- **Connecting Relay:** The IP address of the last recognized relay involved in the transmission of the message.
- **Downstream Relay:** The IP address of the internal server to which the message was routed.
- **Milter Host:** The IP address of the server that is home to the PureMessage mail filter.
- **Message ID:** The message's ID header.
- **Action:** The action performed by PureMessage (for example, "Deliver" or "Quarantine"). If the message is addressed to multiple recipients for which different actions are required, details of how the message was processed for each recipient are shown in the **Activity** section of the **Info** tab.
- **Reason:** The reason that the specified action was performed (for example, "Spam" or "Offensive"). If the message is addressed to multiple recipients for which different actions are required, details of how the message was processed for each recipient are shown in the **Activity** section of the **Info** tab.

Activity

The **Activity** section of the **Info** tab contains a summary of the message-processing actions. If the message is addressed to multiple recipients, the activity for each recipient is shown in its own numbered "Results" section.

Raw Logs

Further details may be available on the **Raw logs** tab, which includes raw text from the mail transfer agent (MTA) log and PureMessage's `message_log`. The **Raw logs** tab can contain one or all of the following sections:

- **MTA Incoming:** The MTA routing actions executed to receive the message.
- **Message Log:** The complete message log entry for this message.
- **MTA Outgoing:** The MTA routing actions executed to deliver the message.

3.4.3 Viewing and Managing Reports

PureMessage offers predefined reports that provide key performance statistics in the form of graphs and tables.

The global administrator has access to the same reports as group administrators and can also access the Queue Status report, which provides statistics about queued messages.

Note

The reporting options described in this section are only available if the Groups Web Interface access rights for these options are enabled. All options are enabled by default. To access the Groups Web Interface, point a supported browser at `https://<Hostname>:28443/groups`. See "Creating a Full-Access Administrator Account" and the Groups Setup tutorial for more information.

Report Types

The names of all the PureMessage reports are displayed on the **Report Types** sidebar of the **Reports** tab. The following reports are available:

Mail Trends

- **Message Categories:** Shows the number of messages detected as blocked, spam, virus, delivered, or other (neither virus nor spam). If PureMessage determines that a message contains spam and also contains a virus, the message counts toward the virus total only. By default, a message is marked as being spam if its spam probability score is 50% or greater.
- **Message Categories per Group:** Shows the number of messages (per group) detected as blocked, spam, virus, delivered, or other (neither virus nor spam).
- **Top Virus Types:** Shows the virus types (categorized by virus ID) found in messages. For details on recent viruses by ID, see the "Latest virus identities" page on the Sophos website.
- **IP Blocked Messages:** Shows the number of messages blocked and not blocked at the policy level on the basis of IP address.
- **Rejected MTA Connections:** Shows the number of connections rejected due to MTA-level IP blocking.
- **Policy Mark Hits:** Shows a count of keys in the message log. If log-marking actions have been added to specific policy rules, this report can be used to monitor the number of times those rules are triggered.
- **Queue Status:** This report is only available to global administrators, and it only applies if you are using Postfix as your mail transfer agent (MTA). Postfix is installed by default with the Full Install

installation option. This report shows the total and top 20 domains' messages queued overall and by each server. The times are shown in minutes. The **Report Parameters** sidebar contains a **Queue** drop-down list for selecting a specific Postfix queue. The **Default** queue displays the combined activity from **Active** and **Incoming** queues.

Senders

- Top Relays: Shows the top relays by number of messages.
- Top Spam Relays: Shows the top spam relays by number of detected spam messages.
- Top Virus Relays: Shows the virus types (categorized by virus ID) found in messages. For details on recent viruses by ID, see the "Latest virus identities" page on the Sophos website.
- Top Other Relays: Shows the top spam relays by number of other messages (those that are classified as neither spam nor virus). This report can help you fine-tune your spam filtering performance by highlighting other relays (for example, partners) you may want to add to your approved hosts.

Recipients

- Top Spam Senders: Shows the top spam senders by number of detected spam messages.
- Top Virus Senders: Shows the top virus senders by number of detected virus messages.
- Top Spam Recipients: Shows the top spam recipients by number of detected spam messages. This report can help you understand which users receive large volumes of spam, allowing you to create custom policy rules to more aggressively filter spam for a group of users or a specific individual.

Note

In addition to these predefined reports, you can create custom policy reports that are also displayed in the Groups Web Interface. Any custom reports are added to a new section at the bottom of the **Report Types** sidebar. For more information, see "Creating a Custom Policy Report" in the Administrative Groups section of the *Administrator's Reference*.

Related concepts

[Tutorial: Groups Setup](#) (page 229)

Related tasks

[Creating a Full-Access Administrator Account](#) (page 207)

Related information

[Latest virus identities](#)

[pmx-reports-custom](#)

Viewing a Report

The **Report Types** sidebar provides access to the individual report pages, where you can define the content and format of the report, and the time period the report covers.

To view a report:

On the **Report Types** sidebar, click the name of the report that you want to view.

The report is displayed in its default format in the content pane, along with the **Report Parameters** sidebar.

Generating a Report

Use the **Report Parameters** sidebar to set the time period, format and content for a report and to generate a report.

To generate a report:

1. Define the report using the options on the **Report Parameters** sidebar:

Note

These options do not apply to the Queue Status report.

- **Period:** Select a predefined time range from the drop-down list (for example, **Last 24 hours** or **Last 30 days**). Each time range in this list refers to complete increments only, and does not include partially elapsed time increments. For instance, **Last 12 hours** defines the 12 most recently complete hours, not including any time that has elapsed in the current hour.

If you select **Today**, the report will cover the period starting at midnight of the current calendar day and ending at the time the report is run.

Alternatively, select **Custom values** if you prefer to specify a custom date range using the text boxes below.

- **Start Date:** Type the date and time in YYYY-MM-DD [hh:mm] format. If you do not specify hours and minutes, the start time is 0:00.
- **End Date:** Type the date and time in YYYY-MM-DD [hh:mm] format. If you do not specify hours and minutes, the end time is 23:59.

Regardless of whether you selected a predefined period or a custom time frame, the report data will be displayed in set increments, according to the length of the period. The following restrictions apply:

- Periods up to 24 hours are shown in hours
 - Periods up to 31 days are shown in days
 - Periods up to one year are shown in months
 - Periods greater than one year are shown in years
 - **Format:** Select the format the report will use to display data (for example, **Bar** or **Table**). The options vary depending on the report, and this option cannot be configured for certain reports.
 - **Traffic:** If applicable, select whether the report will display data for **Inbound** mail, **Outbound** mail, or **Both**.
 - **Groups:** This drop-down list is only displayed if you are a member of multiple groups. Select the specific group for which you want to generate report data, or select **All groups** to generate aggregate data from the various groups.
2. Click **Generate**.

A report with your specified criteria is displayed in the content pane.

Note

If you want reports to be run and sent to a specified email recipient on a regular basis, you can create a scheduled job for `pmx-reports-mailer-v2`. For more information, see the `pmx-reports-mailer-v2` man page, and “Managing Scheduled Jobs” in the *Manager Reference*.

Related tasks

[Managing Scheduled Jobs](#) (page 172)

Related information

[pmx-reports-mailer-v2](#)

Exporting a Report

By default, all reports have an **Export** button that is used to export reports in comma-separated value (CSV) format to other programs for viewing and printing. If a report does not have an **Export** button it is because the global administrator has not made this option available.

To export a report:

1. Click the **Export** button (located beneath the generated report).
2. Use your web browser's file download feature to specify the export location.

Creating a Custom Policy Report

You can add custom reports to the **Reports** page of the Groups Web Interface. Creating a custom report requires several steps, including adding a related action or actions to the PureMessage policy, and registering the report.

Links to custom reports are displayed in the **Policy Reports** section at the bottom of the **Report Types** sidebar. They are based on marks that are added to the message log using the Custom policy mark (`pmx_custom_mark`) action.

Custom reports can be set up with or without the `:overwrite` option (called **Log only one mark per message for the specified key** in the Policy Constructor). Whether you use this option will depend on the nature of the report. Examples of both follow.

Related information

[pmx-reports-custom](#)

Example: Creating a Custom Report

In this example, you will create a custom report that keeps track of outgoing messages that are likely to result in a loss of sensitive data. The report will monitor messages containing the words "Top Secret" or a credit card number. It will also record the number of times a message is sent with an attachment entitled `Internal Financial Report.pdf`.

1. Edit the PureMessage Policy

Marks for custom reports are not logged unless you add an action for each value that you want reported. You can add a policy mark action to an existing rule, or, if necessary, create a new rule. Although this example shows the lines that you must insert in the `policy.siv` file, you can also

modify the policy as necessary using the Policy Constructor. For more information, see “Editing the Policy” in the *Manager Reference*. This custom report requires the following rules:

```
# attr NAME=Log messages containing TOP SECRET keywords
if pmx_phrase :attachments :scanall :contains ["Top Secret"] {
    pmx_custom_mark "dlp_keyword_report" "Top Secret";
}

# attr NAME=Log messages that contain the PDF named Internal Financial
# Report
if pmx_attachment_name :tft :contains ["Internal Financial
Report.pdf"] {
    pmx_custom_mark "dlp_keyword_report" "Financial Reports";
}

# attr NAME=Log messages that contain a Credit Card
    if pmx_credit_card {
pmx_custom_mark "dlp_keyword_report" "Credit Card Numbers";
    }
```

Notice that the last line of each rule contains a `pmx_custom_mark` action.

The key that you enter in the next step must match this key (`dlp_keyword_report`) exactly. Each action has a unique value (for example, "Credit Card Numbers"), which must also exactly match a corresponding entry you will enter in step 2.

2. Register the Report

Now that you have created the necessary logic in the PureMessage policy, you must use the `pmx-reports-custom` tool to register the report. This must be done at the command line, as the PureMessage user ("pmx6" by default).

- a) The values (categories) for the report must be added by way of an external file. At the command line, create a file called `DLP_Report` that includes these values:

```
Top Secret
Financial Reports
Credit Card Numbers
```

- b) Register a report that contains the values shown below. At the command line, run:

```
pmx-reports-custom add --key dlp_keyword_report --title "Data Loss
Prevention" --file /tmp/DLP_Report
```

The `--key` must exactly match the key specified in the corresponding policy marks. The `--title` is the name of the report that will be displayed on the **Reports** page. The `--file` is the path to file containing the report values.

The `--key` that you assign must be lowercase. If not, you are prompted to change it. You are then prompted to run `pmx-profile sync-to-db --force --resource=reports_config`.

If you want to modify the values of the report, edit the file (in this case, `DLP_Report`), and run `pmx-reports-custom update`. For a complete list of commands see the `pmx-reports-custom` man page.

3. Synchronizing Servers with the Database

This step is required to update the database and any edge servers in your deployment:

```
pmx-profile sync-to-db --force --resource=reports_config
```

4. Restart the Mail Filter

To add the changes performed in the previous steps, you must restart the mail filter. At the command line, run:

```
pmx-milter restart
```

5. Test the Report with Mail

To ensure that the custom report is working as expected, process some messages that are designed to trigger `pmx_custom_mark` actions, and then gather new data for the message log.

- a) Pass mail through your PureMessage deployment that will trigger report actions.
- b) At the command line, run:

```
pmx-reports-consume-message-log --v2
```

Note

Depending on when this default scheduled job was last run, it could take several minutes to consume the report data.

- c) Navigate to the **Reports** page in the Groups Web Interface. The custom report is displayed in a new section at the bottom of the **Report Types** sidebar.

6. [Optional] Schedule the Mailing of Custom Reports

Use the `pmx-reports-mailer-v2` command to run and mail custom reports.

```
pmx-reports-mailer-v2 --custom "Data Loss Prevention" --mailto  
me@example.com
```

For more information, see “Managing Scheduled Jobs” in the *Manager Reference*.

Related tasks

[Scheduling a Job](#) (page 174)

Related information

[pmx-reports-custom](#)

Example: Creating a Custom Report (With Overwrite)

In this example, you will create a custom report that uses the `:overwrite` option to specify that only a single mark is written to the message log each time for any given key. You can add `:overwrite` for reports in which you only want one entry written to the report, regardless of how many actions were triggered by a single message.

This report will keep track of the number of messages sent to managers in an organization. In the example organization, there are three levels of managers: executives, middle managers, and team managers. There is a policy list associated with each level of management, and list membership is as follows:

- executives belong to all three policy lists: `executives`, `middle-managers` and `team-managers`
- middle managers are members of: `middle-managers` and `team-managers`
- team managers belong to their list only: `team-managers`

Using **:overwrite** will ensure that the custom report will count each message recipient only once, regardless of whether the recipient belongs to multiple policy lists.

1. Edit the PureMessage Policy

Marks for custom reports are not logged unless you add an action for each value that you want reported. You can add a policy mark action to an existing rule, or, if necessary, create a new rule. Although this example shows the lines that you must insert in the `policy.siv` file, you can also modify the policy as necessary using the Policy Constructor. For more information, see “Editing the Policy” in the *Manager Reference*. This custom report requires the following rules:

```
# attr NAME=Recipient is a team manager
if envelope :memberof "to" "team-managers" {
    pmx_custom_mark :overwrite "list_member" "TeamMan";
}

# attr NAME=Recipient is a middle manager
if envelope :memberof "to" "middle-managers" {
    pmx_custom_mark :overwrite "list_member" "MiddleMan";
}

# attr NAME=Recipient is an executive
if envelope :memberof "to" "executives" {
    pmx_custom_mark :overwrite "list_member" "Exec";
}
```

Notice that the last line of each rule contains a `pmx_custom_mark` action. This example uses the `:overwrite` option, which specifies that only a single mark is written to the message log each time for any given key. The key specified in all cases is `list_member`. In this case, because the action for executives is processed last, any other marks are overwritten if the message is sent to a person who belongs to more than one list.

Note

It is important that you apply `:overwrite` to all `pmx_custom_mark` actions for a specific key, or not at all. Failure to do so will affect the accuracy of the report. For more information, see the `pmx_custom_mark` man page.

The key that you enter in the next step must match this key (`list_member`) exactly. Each action has a unique value (for example, "MiddleMan"), which must also exactly match a corresponding entry you will enter in step 2.

2. Register the Report

Now that you have created the necessary logic in the PureMessage policy, you must use the `pmx-reports-custom` tool to register the report. This must be done at the command line, as the PureMessage user ("pmx6" by default).

- a) The values (categories) for the report must be added by way of an external file. At the command line, create a file called `Mgr_Message_Report` that includes these values:

```
TeamMan
MiddleMan
Exec
```

- b) Register a report that contains the values shown below. At the command line, run:

```
pmx-reports-custom add --key list_member --title "Messages to Management" --file /tmp/Mgr_Message_Report
```

The `--key` must exactly match the key specified in the corresponding policy marks. The `--title` is the name of the report that will be displayed on the **Reports** page. The `--file` is the path to file containing the report values.

The `--key` that you assign must be lowercase. If not, you are prompted to change it. You are then prompted to run `pmx-profile sync-to-db --force --resource=reports_config`.

If you want to modify the values of the report, edit the file (in this case, `Mgr_Message_Report`), and run `pmx-reports-custom update`. For a complete list of commands see the `pmx-reports-custom` man page.

3. Synchronizing Servers with the Database

This step is required to update the database and any edge servers in your deployment:

```
pmx-profile sync-to-db --force --resource=reports_config
```

4. Restart the Mail Filter

To add the changes performed in the previous steps, you must restart the mail filter. At the command line, run:

```
pmx-milter restart
```

5. Test the Report with Mail

To ensure that the custom report is working as expected, process some messages that are designed to trigger `pmx_custom_mark` actions, and then gather new data for the message log.

- a) Pass mail through your PureMessage deployment that will trigger report actions.
- b) At the command line, run:

```
pmx-reports-consume-message-log --v2
```

Note

Depending on when this default scheduled job was last run, it could take several minutes to consume the report data.

- c) Navigate to the **Reports** page in the Groups Web Interface. The custom report is displayed in a new section at the bottom of the **Report Types** sidebar.

6. [Optional] Schedule the Mailing of Custom Reports

Use the `pmx-reports-mailer-v2` command to run and mail custom reports.

```
pmx-reports-mailer-v2 --custom "Messages to Management" --mailto me@example.com
```

For more information, see “Managing Scheduled Jobs” in the *Manager Reference*.

Related tasks

[Scheduling a Job](#) (page 174)

Related information

[pmx-reports-custom](#)

3.4.4 Generating a Self-Signed Certificate for the Groups Web Interface

To make an SSL connection more secure, it is recommended that you generate your own self-signed certificate. The following instructions assume that OpenSSL is installed on the system; see <http://www.openssl.org/> for more information.

To generate a self-signed certificate:

1. At the command line, log in as the PureMessage user (by default "pmx6").
2. Change to the `/opt/pmx6/etc/manager/httpd2/` directory beneath the root PureMessage installation directory.
3. Back up `pmx-cert.cert` and `pmx-cert.pem` by running the following commands:

```
mv pmx-cert.cert backup-pmx-cert.cert
mv pmx-cert.pem backup-pmx-cert.pem
```

4. Generate a new `pmx-cert.pem` file with the following command:

```
pmx-cert --dns=<fully qualified domain name of Groups UI server>
--email ="<Administrator's email address>" --ip=<IP address of Groups
UI server>
--url=http://<fully qualified domain name of Groups UI server>
```

5. Ensure that the `SSLCertificateFile` option in the `/opt/pmx6/etc/manager/httpd2/ssl.conf` file is set to `/opt/pmx6/etc/manager/httpd2/pmx-cert.pem`
6. Restart the HTTP (RPC/UI) service with:

```
pmx-httpd stop; httpd
```

Related tasks

[Configuring SSL](#) (page 154)

Related information

[OpenSSL Website](#)

[pmx-cert](#)

3.4.5 Tutorial: Groups Setup

PureMessage's groups management features allow an organization's global administrator to delegate administrative responsibility for certain groups of recipients or for specific domains. In this example, you will act as the global administrator for a university with two sub-domains, using the groups functionality to create groups and assign administrative roles to a variety of users within the institution. The institution in question will be known as Sophos University.

The university uses the following domains:

- `sophosu.example.com`
- `business.sophosu.example.com`
- `science.sophosu.example.com`

The university has the following administrative roles and access rights:

- Assistant Administrator: Full access rights for all domains
- Business Administrator: Full access rights for business.sophosu.example.com
- Science Administrator: Full access rights for science.sophosu.example.com
- Helpdesk: Rights to allow and block lists for all domains
- Human Resources Administrator: Rights to watch lists for all domains and limited quarantine search rights

To prepare the university to begin administering email with the groups model, the global administrator performs the following task in the order shown:

Important

All of the tasks described below must be performed on the central server (CSM).

Creating Groups

To start, create the groups to which email recipients will belong. In this case, Sophos University requires three groups: one for the primary domain (sophosu.example.com) and one for each of the two subdomains (business.sophosu.example.com and science.sophosu.example.com). You will create these groups from the command line on the central server (CSM). The groups will be named “sophos”, “business” and “science”.

To create the groups:

1. Log on to the central server as the “pmx6” user.
2. Run the following commands:

```
pmx-group --add --group sophos --description "primary domain"
pmx-group --add --group business --description "business domain"
pmx-group --add --group science --description "science domain"
```

The newly created groups are stored in subdirectories of the `/opt/pmx6/etc/members-per-group` directory. When you add a group, single-letter subdirectories are automatically created based on the first two letters of the group name. This is done to ensure that the maximum files per directory limit is not exceeded in cases where organizations have large numbers of groups.

The newly created groups are stored in the following locations:

```
/opt/pmx6/etc/members-per-group/s/o/sophos
/opt/pmx6/etc/members-per-group/b/u/business
/opt/pmx6/etc/members-per-group/s/c/science
```

For additional information, see “Creating a Group”

Related tasks

[Creating a Group](#) (page 204)

Adding Members to Groups

Email recipients are included in groups by adding address or domain information to the group.

When associating email users with groups, it is important that each group contain a unique list of recipients. Adding the same recipient address to more than one group may produce unpredictable results when the PureMessage policy processes the address.

You can add members to groups by either editing the group file directly or by using the `pmx-list` command, which is the method used in this tutorial. Notice that individual addresses are specified for the “sophos” group, while all addresses in the “business” and “science” groups are added by specifying these subdomains.

To add members to groups:

1. At the command line, as the “pmx6” user, run the following command:

```
pmx-profile sync-from-db --resource=members-per-group
```

2. Run these commands to populate the groups with recipient addresses:

```
pmx-list add members-per-group#sophos JaneD@example.com BobS@example.com\
      JackW@example.com AliceC@example.com

pmx-list add members-per-group#business @business.sophosu.example.com

pmx-list add members-per-group#science @science.sophosu.example.com
```

The addresses are added to the respective group files in subdirectories of the `/opt/pmx6/etc/members-per-group` directory.

3. Run `pmx-profile sync-to-db --resource=members-per-group` to synchronize the list data with the database.
4. Run `pmx-makemap --grouplist` to create a CDB list that can be synchronized with the database.

For additional information, see “Adding Members to a Group”

The next step is creating accounts for the university staff who will be responsible for administering the groups.

Related tasks

[Adding Members to a Group](#) (page 205)

Creating Administrator Accounts

In this step you will create the accounts for the various administrators responsible for the groups created and populated in steps 1 and 2.

You will create accounts for the following administrative roles:

- Assistant Administrator (George)
- Business Administrator (Frank)
- Science Administrator (Susan)
- Helpdesk (Jerry)
- Human Resources Administrator (Tanya)

To create the administrator accounts:

At the command line, as the “pmx6” user, run the following commands:

```
pmx-user --add --username GeorgeC --fullname "George Cassidy" --email
GeorgeC@example.com --passphrase sophos
pmx-user --add --username FrankB --fullname "Frank Booth" --email
FrankB@example.com --passphrase sophos
pmx-user --add --username SusanS --fullname "Susan Summers" --email
SusanS@example.com --passphrase sophos
pmx-user --add --username JerryS --fullname "Jerry Sanders" --email
JerryS@example.com --passphrase sophos
pmx-user --add --username TanyaH --fullname "Tanya Harrison" --email
TanyaH@example.com --passphrase sophos
```

A confirmation message is displayed for each user added.

For additional information, see “Creating an Administrator Account”.

With groups and administrator accounts created, the next step is associating administrators with their appropriate group(s).

Related tasks

[Creating an Administrator Account](#) (page 206)

Adding Administrator Accounts to Groups

Access rights are granted on the basis of group/administrator pairs. In this step you will create associations between the groups and administrator accounts set up in the previous steps. After you run each of the commands shown below, a message is displayed indicating that the user can administer the specified group.

You will create the following associations:

Role	Username	Associated Groups
Assistant Administrator	GeorgeC	sophos, business, science
Business Administrator	FrankB	business
Science Administrator	SusanS	science
Helpdesk	JerryS	sophos, business, science
Human Resources Administrator	TanyaH	sophos, business, science

To add the administrator accounts to groups:

At the command line, as the “pmx6” user, run the following commands:

```
pmx-group --add-admin --group sophos --user GeorgeC
pmx-group --add-admin --group business --user GeorgeC
pmx-group --add-admin --group science --user GeorgeC

pmx-group --add-admin --group business --user FrankB
pmx-group --add-admin --group science --user SusanS

pmx-group --add-admin --group sophos --user JerryS
pmx-group --add-admin --group business --user JerryS
pmx-group --add-admin --group science --user JerryS

pmx-group --add-admin --group sophos --user TanyaH
pmx-group --add-admin --group business --user TanyaH
pmx-group --add-admin --group science --user TanyaH
```

For additional information, see “Adding an Administrator Account to a Group”

Related tasks

[Adding an Administrator Account to a Group](#) (page 208)

Creating a Custom Group Policy List and Rule

Although the **Configuration** tab of the Groups Web Interface contains several default lists that can be used to determine how the policy deals with specified email addresses and domains, you can also create custom lists.

Note

Regardless of whether the list is a default list or a custom list, it must be combined with a rule in the PureMessage policy, as described in the second part of this procedure.

In this tutorial step you will create an "offensive" list that the policy will use to quarantine messages that contain keywords specified in this list. Then you will use the PureMessage Manager to configure a group-specific policy rule for messages containing offensive words. These messages will be quarantined with the reason "offensive". Finally, you will add an “offensive” option to the quarantine search criteria that will allow group administrators to search for messages that were quarantined for this reason.

Note

This step and steps 6 and 7 require that you edit the policy script using the policy constructor in the PureMessage Manager. It is recommended that you first create a backup of the existing policy that you can revert to when you have finished this tutorial.

First you must back up the policy. On the **Policy** tab of the Manager, next to Backups on the sidebar, click **Create**. A backup entry with the date and time is displayed.

- To create the custom list:

- a) At the command line, as the "pmx6" user, run the following:

```
pmx-group-list --add --id offensive-words-per-group --name
"Offensive Words (per group)" \
--description "Offensive Words" --match-type contains
```

- b) Run `pmx-profile sync-from-db`.

A list with the assigned name is added to `/opt/pmx6/etc` and list data is added to `etc/multilists.conf`. If you use `pmx-group` with the `--view-perm` option, you will see that this list now appears among the permissions that can be enabled and disabled.

- c) For each server that is running the group management web interface, at the command line, run:

```
pmx-httpd restart
```

The list will be accessible the next time the group administrator views the **Configuration** tab.

- To create the associated policy rule:

- a) In the PureMessage Manager, click the **Policy** tab.

The default PureMessage policy is displayed in constructor mode.

- b) At the bottom left of the page, click **add main rule**.

A set of controls for creating a new rule is displayed.

- c) Use the text box and drop-down lists to create a test that matches the one below. Then click **add action** and select the drop-down list options shown to complete the new rule.

- d) Click **Save**.

- e) Click **Cut**.

- f) Click the existing policy rule, **Quarantine mail containing suspicious attachments**.

The details for this rule are displayed.

- g) Click **Paste**.

The Check for offensive content (per group) rule is added beneath the existing rule.

- h) Click **Save**.

- To create the "offensive" quarantine search option:

- a) At the command line, as the “pmx6” user, run the following:

```
pmx-group --add-perm --permission quarantine.reason.offensive
```

A message is displayed advising that the permission has been added. The **Offensive** reason will now appear on the **Reason** drop-down list on the **Search Parameters** sidebar of the **Search** tab.

For more information, see “Creating a Group List”, “Adding and Deleting Custom Reasons”, the `pmx-group-list` man page, and the `pmx-group` man page.

For more information about configuring the PureMessage policy, see “Policy Configuration” in the *Administrator's Reference* and “Policy” in the *Manager Reference*.

Next, you will use `pmx-group` command to add a group policy setting, and then associate the setting with a group-specific policy rule.

Related concepts

[Policy Configuration](#) (page 242)

This section describes the customization and fine tuning of message filtering for viruses, spam, and organizational policy compliance.

[Policy Tab](#) (page 78)

Related tasks

[Creating a Group List](#) (page 210)

[Adding and Deleting Custom Quarantine Search Reasons](#) (page 211)

[Adding and Deleting Custom Log Search Reasons](#) (page 212)

Related information

[pmx-group](#)

[pmx-group-list](#)

Adding and Defining a Policy Setting

By default, the **Policy: Policy Settings** page of the **Configuration** tab has no settings configured. Implementing a group policy option requires two steps. First, you will create a new policy setting, in the form of a check box, that gives group administrators the option of enabling/disabling spam checking. Second, you will use the PureMessage policy constructor to define a new policy rule and associate it with this setting.

- To create the policy setting:
 - At the command line, as the “pmx6” user, run the following:

```
pmx-group-policy --add --id group-anti-spam-opt-outs --name "Anti-
spam opt-outs (per group)" \
  --type checkbox --style optin --description "Disable spam
checking"
```

- Run `pmx-profile sync-from-db`.

A check box option, **Disable spam checking**, is now displayed on the **Policy: Policy Settings** page of the **Configuration** tab. In addition, a new group list, **Anti-spam opt-outs (per group)**, is added to `/opt/pmx6/etc/lists.conf` and is also available in the drop-down list that is used for specifying lists in the PureMessage policy constructor.

-
- To define the policy rule:
 - In the PureMessage Manager, click the **Policy** tab.

- The default PureMessage policy is displayed in constructor mode.
- b) Click the existing policy rule, **Deliver mail to anti-spam opt-outs**.
- The details for this rule are displayed.
- c) Click **add test**.
- Additional **Tests** drop-down lists are displayed.
- d) Select the drop-down list options as shown below.

The screenshot shows the configuration window for the policy rule "Deliver mail to anti-spam opt-outs". The interface includes a "Tests" section with two tests: "Envelope to" and "Envelope group". Each test is configured with the condition "Is a member of" and the target "Anti-spam opt-outs" (for the first test) and "Anti-spam opt-outs (per group)" (for the second test). A summary box indicates "If ANY criteria are met". Below the tests, the "Execute actions and rules" section shows two actions: "Accept the message" and "Stop processing". At the bottom, there are buttons for "Save", "Cancel", "Copy", "Cut", "Delete", and "Add Alternative".

- e) Click **Save**.

For additional information, see "Creating a Policy Setting", and the `pmx-group-policy` man page.

For more information about configuring the PureMessage policy, see "Policy Configuration" in the *Administrator's Reference* and "Policy" in the *Manager Reference*.

Next, you will learn how to configure a group-specific document that can be viewed by global administrators.

Related concepts

[Policy Configuration](#) (page 242)

This section describes the customization and fine tuning of message filtering for viruses, spam, and organizational policy compliance.

[Policy Tab](#) (page 78)

Related tasks

[Creating a Policy Setting](#) (page 209)

Related information

[pmx-group-policy](#)

Adding a Group Document

Sometimes you may want group administrators to have access to relevant custom documentation. Although you can choose to permit group administrators to edit these documents, document templates are generally read-only. In this tutorial step you will create a document template using the `pmx-group-file` command, and add content via the group administrator interface. Permissions for the template will be set in step 9.

- To add a group document template:
 - At the command line, as the “pmx6” user, run the following:

```
pmx-group-file --add --id policy-description --name "Policy
Description"\
--type document
```

- Run `pmx-profile sync-from-db --all`.
- Run `pmx-httpd restart`.

A new template is added to the `/opt/pmx6/var/lib/multifile` directory. In addition, a new category, Documents, is displayed on the sidebar of the **Configuration** tab in the Groups Web Interface.

- To add content to the document:
 - Log in to the Groups Web Interface: point a supported browser at `https://<Hostname>:28443/groups`, and log in as one of the users you created in step 3 (for example, Username: GeorgeC Password: sophos). Accept the certificate to continue to the GUI.

The GUI launches with the **Search** tab displayed.

- Click the **Configuration** tab.

On the **Configuration** sidebar, the **Policy Description** link is displayed beneath the **Documents** heading.

- Click **Policy Description**.

The **Documentation** template is displayed in the content pane.

- Copy and paste the content below into the text box provided, and click **Save**.

```
Mail from internal hosts
  Reject mail containing viruses
    Allow unscannable messages to pass through

Mail from external hosts
  Clean mail containing viruses
  Quarantine mail containing suspicious attachments
  Deliver mail from whitelisted hosts and senders
  Deliver mail to anti-spam opt-outs
  Quarantine mail from blacklisted hosts and senders
  Copy to quarantine and deliver if spam probability is 50% or
more
  Add X-Header and deliver messages
```

A message is displayed at the bottom of the content pane, indicating that the Policy Description has been saved. The access rights for this document are currently read/write, but you will change the status to read-only as part of step 9, "Granting Access Rights."

For additional information, see "Creating a Policy Setting" and the `pmx-group-file` man page.

Next, you will see how to add a group-specific banner.

Related tasks

[Creating a Group List](#) (page 210)

Related information

[pmx-group-file](#)

Adding a Group Banner

In this step you will create a group-specific add banner option that will be available to group administrators. First you will use the `pmx-group-file` command to create the group-specific banner option. Then you will edit the PureMessage policy to create an action that can be used to append a disclaimer to all outgoing mail.

- To add a group banner:
 - a) At the command line, as the "pmx6" user, run the following:

```
pmx-group-file --add --id outgoing-disclaimer-per-group --name
  "Outgoing Disclaimer" \
  --type banner --charset latin1
```

- b) Run `pmx-profile sync-from-db --all`.
- c) Run `pmx-httpd restart`.

A new template is added to the `/opt/pmx6/var/lib/multifile` directory. In addition, a new category, Banners, is displayed on the sidebar of the **Configuration** tab. The group administrator can now click **Outgoing Disclaimer** to display a text box for adding banner text.

- To create a group-specific "add banner" action:
 - a) In the PureMessage Manager, click the **Policy** tab.

The default PureMessage policy is displayed in constructor mode.

- b) At the bottom left of the page, click **add main rule**.

A set of controls for creating a new rule is displayed.

- c) Use the text box and drop-down lists to create a rule identical to the one shown below.

Add group-specific outgoing disclaimer

Tests: [add test](#)

Always match

Execute actions and rules: [add action](#) [add rule](#)

Add banner

Arguments...

Save Cancel Copy Cut Delete Add Alternative

- d) Click the **Arguments** button.

The **Add banner** dialog box is displayed.

- e) In the **Add banner** dialog box, select **Group-specific banner**. In the adjacent text box, type `outgoing-disclaimer-per-group`, which is the ID you assigned to this banner template using the `pmx-group-file` command.
- f) Click **OK**.
- g) Click **Save**.
- h) Click **Cut**.
- i) Click the existing rule, **Reject mail containing viruses**, in the Mail from internal hosts section of the policy.
- j) Click **Paste**.
- k) Click **Save**.
- l) Click **Commit**.

For additional information, see [Creating a Policy Setting](#) and the `pmx-group-file` man page.

Next, you will see how to grant access rights to the various administrators.

Related tasks

[Creating a Group List](#) (page 210)

Related information

[pmx-group-file](#)

Granting Access Rights

Access rights are set on the basis of group/administrator pairs. By default, any group that has been associated with an administrator account has full access rights enabled. In this step you will change the permissions as necessary, so that some of the administrators will only be able to access certain tabs and options in the Groups Web Interface.

The access rights will be granted as follows:

Role	Username	Access Rights
Assistant Administrator	GeorgeC	Full access to all domains.
Business Administrator	FrankB	Full access within the "business" domain.
Science Administrator	SusanS	Full access within the "science" domain.
Helpdesk	JerryS	Allow and block lists for all domains (no other Configuration options), online help access, quarantine (with no preview options) and no access to reports.
Human Resources Administrator	TanyaH	"Offensive Words" watch list for all domains, online help access, quarantine access (for reason "offensive" only), and no access to reports.

In this tutorial, the Assistant Administrator (GeorgeC) has responsibility for all of the domains. Since full access rights were granted by default when you associated this user with each of the three groups, there is no need to modify the permissions for GeorgeC. The same is true for FrankB (Business Administrator) and SusanS (Science Administrator), who already have full access to their respective domains.

For rest of the administrators, however, you will have to restrict access to certain features. This is accomplished by specifying the --group, --user, --permission and --value (usually "on" or "off") for specific permissions or groups of permissions.

To set permissions for the Helpdesk and Human Resources administrators:

At the command line, as the "pmx6" user, run the following commands:

Helpdesk - JerryS

```
pmx-group --set-perm --group sophos --user JerryS --permission
configuration.document.policy-description --value read
pmx-group --set-perm --group sophos --user JerryS --permission
configuration.policysettings --value off
pmx-group --set-perm --group sophos --user JerryS --permission
quarantine.preview --value off
pmx-group --set-perm --group sophos --user JerryS --permission reports --
value off

pmx-group --set-perm --group business --user JerryS --permission
configuration.document.policy-description --value read
pmx-group --set-perm --group business --user JerryS --permission
configuration.policysettings --value off
pmx-group --set-perm --group business --user JerryS --permission
quarantine.preview --value off
pmx-group --set-perm --group business --user JerryS --permission reports
--value off

pmx-group --set-perm --group science --user JerryS --permission
configuration.document.policy-description --value read
pmx-group --set-perm --group science --user JerryS --permission
configuration.policysettings --value off
pmx-group --set-perm --group science --user JerryS --permission
quarantine.preview --value off
pmx-group --set-perm --group science --user JerryS --permission reports
--value off
```

Human Resources Administrator - TanyaH

```

pmx-group --set-perm --group sophos --user TanyaH --permission
configuration.document.policy-description --value read
pmx-group --set-perm --group sophos --user TanyaH --permission
configuration.lists.allowed-relays-per-group --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
configuration.lists.allowed-senders-per-group --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
configuration.lists.blocked-relays-per-group --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
configuration.lists.blocked-senders-per-group --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
configuration.policysettings --value off
pmx-group --set-perm --group sophos --user TanyaH --permission help --
value off
pmx-group --set-perm --group sophos --user TanyaH --permission
quarantine.actions --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
quarantine.actions.approve --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
quarantine.actions.delete --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
quarantine.actions.forward --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
quarantine.actions.report --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
quarantine.actions.save --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
quarantine.preview.attachments --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
quarantine.preview.attachments.download --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
quarantine.preview.content --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
quarantine.preview.info --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
quarantine.preview.source --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
quarantine.preview.status --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
quarantine.reason.blacklisted --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
quarantine.reason.spam --value off
pmx-group --set-perm --group sophos --user TanyaH --permission
quarantine.reason.virus --value off
pmx-group --set-perm --group sophos --user TanyaH --permission reports --
value off

pmx-group --set-perm --group business --user TanyaH --permission
configuration.document.policy-description --value read
pmx-group --set-perm --group business --user TanyaH --permission
configuration.lists.allowed-relays-per-group --value off
pmx-group --set-perm --group business --user TanyaH --permission
configuration.lists.allowed-senders-per-group --value off
pmx-group --set-perm --group business --user TanyaH --permission
configuration.lists.blocked-relays-per-group --value off
pmx-group --set-perm --group business --user TanyaH --permission
configuration.lists.blocked-senders-per-group --value off
pmx-group --set-perm --group business --user TanyaH --permission
configuration.policysettings --value off

```

The permissions are disabled for the specified users.

The `pmx-group` command is also used to view permissions for a specific group/administrator pair. For example, you can view the complete list of permissions that the user “TanyaH” has for the “business” group by running the following command:

```
pmx-group --view-perm --group business --user TanyaH
```

For additional information, see “Setting Group Access Rights” and “Viewing Group Access Rights”.

You have completed the tutorial. The groups you created can now be administered according to the roles and permissions you defined.

Related concepts

[Viewing Group Access Rights](#) (page 209)

Related tasks

[Setting Group Access Rights](#) (page 208)

3.5 Policy Configuration

This section describes the customization and fine tuning of message filtering for viruses, spam, and organizational policy compliance.

Policies, lists and maps provide the means to configure the way in which PureMessage filters email. Policy configuration determines how messages containing viruses or spam messages are handled PureMessage, as well as allowing you to filter messages for certain keywords. Lists include domains that either generate mostly spam or that try to remove all spam. Maps are simple tables of email address rerouting, often used to send messages addressed to several different email accounts to one actively used account.

This section describes how the default PureMessage policy works and how it can be customized. It provides details on the PureMessage policy spam detection and virus handling settings, and it describes how to modify the policy using either the Manager or the command-line interface.

3.5.1 Policy Overview

Policies provide the filtering definition for your PureMessage installation. Policies consist of rules; rules consist of tests and actions. As messages pass through the policy, rules are executed on the message in the order of their configuration. Each rule can have a “stop” action, which prevents the message from being processed any further.

Configure policies on the **Policy** tab of the PureMessage Manager, or by using the `pmx-policy` command-line program. Before editing a policy, it is strongly recommended that you back up the current policy.

Note

The `policy.siv` script is never automatically updated, but you can update it manually. If you want to update a modified `policy.siv` script, first back it up, then run the following command:

```
pmx-policy integ --regen
```

This command generates a new default policy based on the latest policy template. To restore your modifications, you should run a diff between the backed up, modified version and the regenerated version, and you must then manually paste in the changes that you want to preserve.

Related concepts

[Policy Tab](#) (page 78)

[Backing Up and Reverting Policies](#) (page 270)

Related information

[pmx-policy](#)

The Default PureMessage Policy

A default policy is installed and enabled during the PureMessage installation. The default policy varies according to your PureMessage license; for example, if you do not have a license for the PureMessage Virus component, virus-checking rules are not configured.

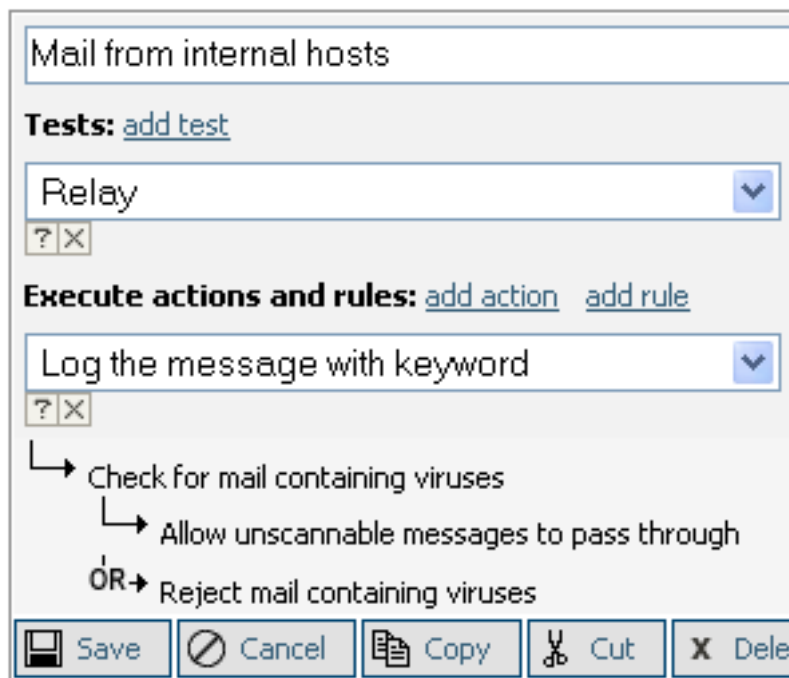
In this section, the default policy is examined in four parts. Each part describes a sub-set of the rules and actions within the default policy.

Related concepts

[Default PureMessage Policy Script](#) (page 250)

Part One: Internal Hosts

The first part of the policy script handles messages from internal hosts. The Manager screen shot indicates where to edit internal host rules and actions within PureMessage. A step-by-step description of this part of the policy is provided below.



Description:

Before PureMessage runs its tests and actions for internal hosts, a mark is added to the message log for both incoming and outgoing messages so that it is possible to search by subject when using the log search feature of the Groups Web Interface.

- If the message originated from a relay defined in the Internal Hosts list:
 - A mark is added to the message log to enable Perimeter Protection to distinguish outgoing messages from internal hosts.
- If the message contains unscannable data:
 - A header is added to the message indicating that the message could not be scanned and that it may contain a virus.
 - A mark is added to the message log indicating that the message was unscannable.
 - Message processing stops.
- If the message contains a virus:
 - The message is rejected with the reason "One or more viruses were detected in the message".
 - Message processing stops.

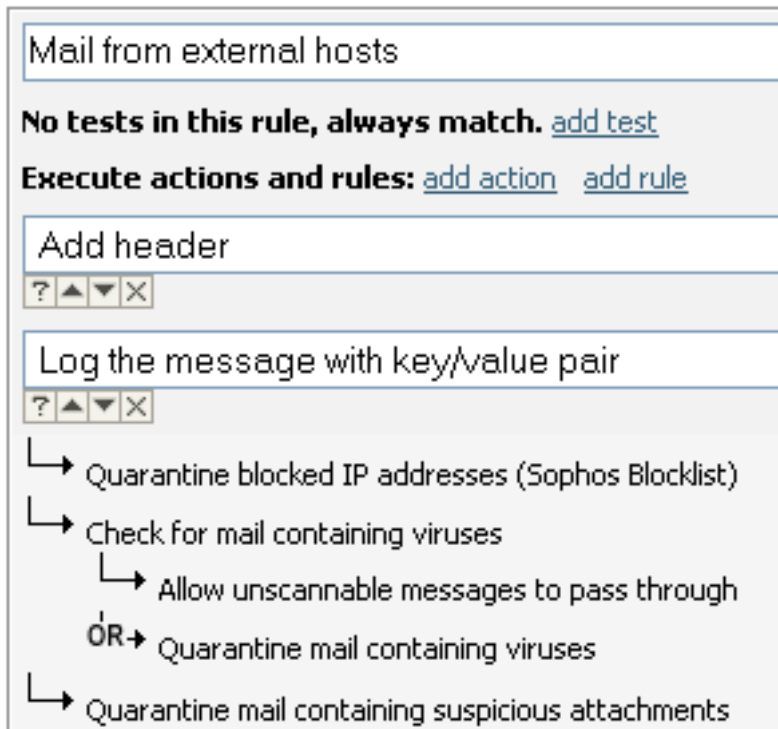
Related concepts

[Lists](#) (page 252)

[Viruses](#) (page 278)

Part Two: External Hosts

The second part of the policy script handles messages from external hosts. Depending on message content, this part of the policy script scans for viruses and suspect attachments. The Manager screen shot shows where to edit virus-checking rules and actions for messages from external hosts. A step-by-step description of this part of the policy is provided below.



Description:

Messages not originating from a relay defined in the Internal Hosts list are assumed to be from an external host. Messages from external hosts are scanned for viruses and suspect attachments in this part of the policy script, as follows:

- A header is added to the message (X-PMX-Version and the PureMessage version number).
- The size of the message is written to the message log.
 1. If the message contains an IP address that belongs to the Sophos blocklist:
 - The message is quarantined and a mark is added to the message log indicating that the message contained a blocklisted IP address.
 2. If the message cannot be scanned:
 - Text is added to the subject of the message indicating that message was not scanned and that it may contain a virus.
 - A mark is added to the message log indicating that the message is unscannable.
 3. If the message contains a virus:
 - A copy of the message is written to the quarantine with the reason "Virus".
 - A mark is added to the message log indicating that the message contains a virus.
 4. If the message contains a suspicious attachment:
 - The message is sent to the quarantine with the reason "Suspect".
 - A mark is added to the message log indicating that the message contains a suspect attachment.
 - Message processing stops.

Related concepts

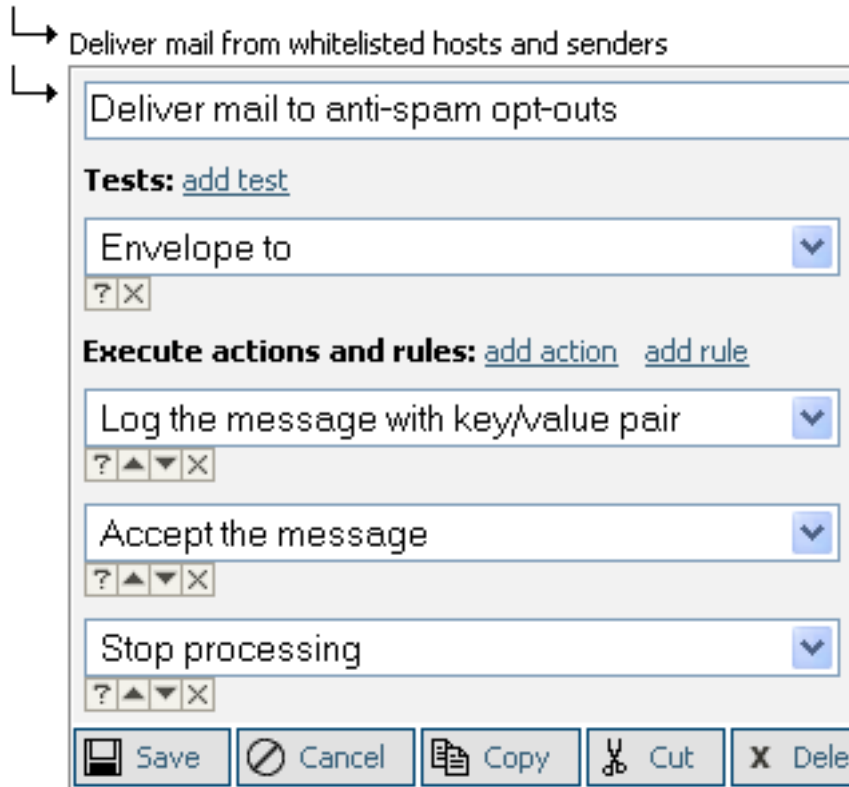
[Lists](#) (page 252)

[Viruses](#) (page 278)

[About Tests](#) (page 81)

Part Three: External Hosts, Lists

The third part of the policy script continues to handle messages from external hosts previously filtered in Part Two: External Hosts. Specifically, this part of the policy filters messages on whitelisted relays, anti-spam opt-out lists, and blacklisted relays. The Manager screen shot shows where to edit list-checking rules and actions for messages from external hosts. A step-by-step description of this part of the policy is provided below.



Description:

Messages reaching this part of the policy script are from external hosts and have been previously scanned for viruses and suspect attachments. These contagion-free messages are now checked for membership on whitelists, blacklists, and anti-spam opt-out lists, as follows:

1. If the message originates from a Relay defined in the Whitelisted Hosts list or the message has an Envelope From address defined in the Whitelisted Senders list or Whitelisted Senders (per-user) list:
 - The message is delivered.
 - A mark is added to the message log indicating that the message originated from a whitelisted address.
 - Message processing stops.
2. If the message recipient address is defined in the Anti-Spam Opt-Out list:
 - The message is delivered.
 - A mark is added to the message log indicating that the recipient is a member of the Anti-Spam Opt-Out list.

- Message processing continues for recipients who are not defined in the Anti-Spam Opt-Out list. (If all recipients are included in the list, message processing stops.)
3. If the message originates from a Relay defined in the Blacklisted Hosts list or the message has an Envelope From address defined in the Blacklisted Senders list or Blacklisted Senders (per-user) list:
- The message is quarantined with the reason "Blacklisted".
 - A mark is added to the message log indicating that the message originated from a blacklisted address.
 - Message processing stops.

Related concepts

[About PureMessage Default Lists](#) (page 118)

[Managing End User Lists](#) (page 139)

[Lists](#) (page 252)

Part Four: External Hosts, Spam Probabilities

The fourth part of the policy script applies spam probabilities to those messages that have passed through the third part of the default policy filter, (External Hosts, Lists). The Manager screen shot shows where to edit spam probability rules and actions for messages from external hosts. A step-by-step description of this part of the policy is provided below.

L → Copy to quarantine and deliver if spam probability is 50% or more
 OR →

Add X-Header and deliver messages
 No tests in this rule, always match. [add test](#)
 Execute actions and rules: [add action](#) [add rule](#)

Log the message with key/value pair
 ? ▲ ▼ X

Replace header
 ? ▲ ▼ X

Stop processing
 ? ▲ ▼ X

Save Cancel Copy Cut Delete

Description:

Messages reaching this part of the policy script are from external hosts and have been previously scanned for viruses and membership on various PureMessage lists (whitelists, blacklists, anti-spam opt-outs). Messages only reach the 'spam Probabilities' section of the policy script because they haven't met the requirements for quarantining or sending. These messages now undergo specific scans testing for spam qualities, as follows:

1. If the spam probability for the message is 50% or more:

- An `X-PerlMx-Spam` header is added (or altered, if it already exists) that contains details of the scan, including the spam probability and the anti-spam rules violated by the message.
 - A copy of the message is written to the quarantine with the reason "Spam".
 - A mark is added to the message log indicating that the message is considered to be spam .
 - The subject header in the message is prefixed by the text `PMX:` and a `#` symbol for every 10% that the message's spam probability exceeds 50%.
 - The message is delivered.
 - Message processing stops.
2. Otherwise:
- An `X-PerlMx-Spam` header is added (or altered, if it already exists) that contains details of the scan, including the spam probability and the anti-spam rules violated by the message.
 - The message is delivered.
 - A mark is added to the message log indicating that the message indicating that it is legitimate mail.
 - Message processing stops.

Related concepts

[Spam Detection](#) (page 271)

[Quarantine Administration](#) (page 280)

This section describes management of the PureMessage quarantine, a temporary holding place for messages that are deemed potentially problematic by the PureMessage policy. Quarantined messages can then be reviewed, and released or deleted.

Default PureMessage Policy Script

This is the default policy enabled for PureMessage. To view the [Sieve](#) code, click **see the source** on the **Policy** tab of the PureMessage Manager. To edit the Sieve code directly, open the `policy.siv` file found in `/opt/pmx6/etc`.

```
require "PureMessage";

# The 'pmx-test-mark' command is needed for the sample messages sent
# by the pmx-test program to be recognized. For sites running with
# high-mail volumes it might be a good idea to disable this action as
# it prevents the relay tests from running as early as they otherwise
# could. See 'perldoc pmx-policy' for details about this command.
pmx_test_mark;
# Mark the subject (for both incoming and outgoing messages)
pmx_mark "S" "%%SUBJECT:h_utf8%%";
# attr NAME=Mail from internal hosts
if pmx_relay :memberof "internal-hosts" {
    # The 'pmx-mlog-watch' depends on this to know which messages
    # are outgoing and which are not.
    pmx_mark1 "i";
    # attr NAME=Check for mail containing viruses
    if pmx_virus {
        # attr LICENSE=PureMessage::Policy::Virus
        # attr NAME=Allow unscannable messages to pass through
        if pmx_virus_cantscan {
            pmx_replace_header :index 0 "X-PMX-Virus" "Unscannable";
            pmx_replace_header :index 0 "Subject" "[POTENTIAL VIRUS] %
%SUBJECT%%";
            pmx_mark "pmx_reason" "Unscannable";
        }
        # attr NAME=Reject mail containing viruses
        else {
            pmx_mark "pmx_reason" "Virus";
            reject "One or more viruses (%%VIRUS_IDS%%) were detected in
the message.";
            stop;
        }
    }
}
# attr NAME=Mail from external hosts
else {
    pmx_add_header "X-PMX-Version" "%%PMX_VERSION%%";
    pmx_mark "Size" "%%MESSAGE_SIZE%%";
    # attr NAME=Quarantine blocked IP addresses (Sophos Blocklist)
    if pmx_blocklist {
        pmx_mark "pmx_reason" "Block List";
        pmx_quarantine "Blocked";
        stop;
    }
    # attr NAME=Check for mail containing viruses
    if pmx_virus {
        # attr LICENSE=PureMessage::Policy::Virus
        # attr NAME=Allow unscannable messages to pass through
        if pmx_virus_cantscan {
            pmx_replace_header :index 0 "X-PMX-Virus" "Unscannable";
            pmx_replace_header :index 0 "Subject" "[POTENTIAL VIRUS] %
%SUBJECT%%";
            pmx_mark "pmx_reason" "Unscannable";
        }
        # attr NAME=Quarantine mail containing viruses
        else {
```

Policy Rules

Policies consist of rules; rules consist of tests and actions.

Lists are used to configure domains or addresses that should be treated differently within a policy rule.

Address maps are used to associate one email address with another, either for the purpose of redirecting notifications, such as Quarantine Digests, or for the purpose of assigning one user's email preferences to other accounts, as it is with the End User Web Interface.

The applications and configuration files used to configure lists and maps from the command line are:

- `/opt/pmx6/bin/pmx-list` : Utility for manipulating lists.
- `/opt/pmx6/bin/pmx-ldap-sync` : Creates a list or map from an LDAP service.
- `/opt/pmx6/bin/pmx-makemap` : Compiles a standard list or map, or a list of members of a PureMessage administrative group, into a CDB list or map.

Lists & Maps Related Configuration Files

- `/opt/pmx6/etc/lists.conf` : Declares the named predefined lists available PureMessage.
- `/opt/pmx6/etc/multilists.conf` : Declares the named multidimensional lists available to PureMessage.
- `/opt/pmx6/etc/maps.conf` : Declares the named predefined maps available to PureMessage.

Related information

[pmx-list](#)
[pmx-ldap-sync](#)
[pmx-makemap](#)
[lists.conf](#)
[multilists.conf](#)
[maps.conf](#)

Tests

Tests define the characteristics of the message that must be matched in order for the action to be executed. Multiple tests can be configured within a single rule. See the "About Tests" in the *Manager Reference*, or the `pmx-policy` man page, for a list of tests. To see which modules are licensed on your system, see the **View Licensed Components** page on the **Support** tab of the PureMessage Manager.

The components of a test are:

- **Message Characteristic:** Specifies the component of the message that is being tested, such as the number of attachments or the percentage of 8-bit characters.
- **Operator:** Defines the comparison between the specified message characteristic and the specified test expression. For example, if the message characteristic being analyzed is the number of attachments, the operator specifies a numerical test, such as "Is over" or "Is under".
- **Test Expressions:** Specifies the value that is compared to the message characteristic.

Related concepts

[About Tests](#) (page 81)
[Support Tab](#) (page 189)

Related information

[pmx-policy](#)

Actions

Actions are the components of rules that determine what happens to a message that passes the test. Multiple actions can be specified. See “About Actions” in the *Manager Reference*, or the `pmx-policy` man page for a list of actions. To see which modules are licensed on your system, see the **View Licensed Components** page on the **Support** tab in the PureMessage Manager.

Related concepts

[Tests](#) (page 251)

[About Actions](#) (page 93)

[Support Tab](#) (page 189)

Related information

[pmx-policy](#)

Lists

Lists are used to configure domains or addresses that should be treated differently within a policy rule. For example, in the default PureMessage policy, the first rule checks the message relay against the relays configured in the Internal Hosts list.

To handle messages differently for members in each list, add a new list, and then apply it within the policy script. For example, to create a list of users who receive spam headers for messages with a spam rating of 50% or more, build a custom list (or use an existing list), and then create a rule that adds a header only if the recipient is a member of the list.

Some message characteristics, such as “Recipient’s address” and “Envelope to”, provide the option of using the match operators “Is a member of” and “Is not a member of” to compare the message characteristic with items in a specified list. Once the new list is created, it becomes available for selection in a Policy Constructor drop-down list that is displayed when you are configuring a test that includes the “Is a member of” or “Is not a member of” match operator.

Note

Every item in a list is evaluated independently, so the order of the items within a list does not matter.

Note

Although PureMessage supports the creation of LDAP-based lists and maps, these lists and maps are read-only; you must use LDAP tools to edit them.

Related concepts

[Message Characteristics](#) (page 81)

[Operators](#) (page 91)

[About PureMessage Default Lists](#) (page 118)

Related tasks

[Creating Lists or Maps](#) (page 114)

[Testing Lists or Maps](#) (page 121)

Related information

[lists.conf](#)

Address Maps

Address maps are used to associate one email address with another, either for the purpose of redirecting notifications generated by PureMessage (such as Quarantine Digests), or for the purpose of assigning one user's email preferences to other accounts (for End User Web Interface usage).

The default address maps (the **Notifications Address Map** and the **Recipient Aliases Map**) are implemented automatically; they are not explicitly implemented in the policy script. To use these maps, simply populate them with the desired values. Custom address maps must be implemented via a policy rule.

PureMessage processes the contents of address maps from top to bottom. Within a map, if two mappings apply to a single address, the first mapping is used. For example, if the first entry maps `sales*@example.com` to `joe@example.com`, and the second entry maps `saleslocal@example.com` to `mary@example.com`, all messages with addresses beginning with "sales" are mapped to Joe, not to Mary.

PureMessage is distributed with two address maps:

- **Notifications Address Map:** Accessible from the **Quarantine** or **Policy** tab in the PureMessage Manager, or by editing the `notifications` file, located in the `etc` directory beneath the root PureMessage installation directory.

Note

The `/opt/pmx6/etc/notifications` file is a shared resource. If you edit this file, you must sync it to the database with the following command (run as the PureMessage user):

```
pmx-profile sync-to-db --resource=notifications --force
```

The Notifications Address Map redirects PureMessage notifications (such as quarantine digests and virus notices) from the original message recipient to the recipient specified in the map. For example, the address `sales@example.com` might be administered by a user called `joe@example.com`. An entry in the Notifications Address Map that maps `sales@example.com` to `joe@example.com` ensures that messages generated by PureMessage to the address `sales@example.com` are sent to `joe@example.com`. (Note that the Notifications Address Map does not consolidate digests; see **Consolidating Quarantine Digests** in the Quarantine Manager chapter for instructions on that feature.)

- **Recipient Aliases Map:** Accessible from the **Policy** tab in the PureMessage Manager, or by editing the `recipient-aliases` file, located in the `etc` directory beneath the root PureMessage installation directory.

Note

The `/opt/pmx6/etc/recipient-aliases` file is a shared resource. If you edit this file, you must sync it to the database with the following command (run as the PureMessage user):

```
pmx-profile sync-to-db --resource=recipient-aliases --force
```

The Recipient Aliases Map replaces the original message recipient with another recipient for the purpose of applying user preferences (such as per-user whitelists and blacklists, as well as viewing messages in the **End User Web Interface**). For example, if the address `feedback@example.com` is administered by the user `karen@example.com`, the recipient alias map could be used to assign the preferences associated with `karen@example.com` to the address `feedback@example.com`. While this does not alter the actual recipient address (that

is, the message is delivered to `feedback@example.com`), the PureMessage policy uses the preferences set for `karen@example.com` while processing the message.

When recipient alias mapping is enabled for a user, per-user whitelist and blacklist entries made via the **End User Web Interface** (or via the **End User Whitelist** and **End User Blacklist** options on the **Quarantine** tab of the PureMessage Manager), they are only applied for the destination of the recipient alias map, not the source. For example, if `feedback@example.com` is mapped to `karen@example.com`. End User Blacklist and Whitelist entries for `feedback@example.com` are ignored; instead, user preferences for `karen@example.com` are applied, and `karen@example.com` will also be able to view the messages for `feedback@example.com` in the **End User Web Interface**.

For information about populating map contents, or for information about configuring custom maps, see “Creating Lists or Maps”. Also see the “Operators” and “Wildcard Usage” sections for information on matching email addresses, hostnames and IP addresses.

Related concepts

[End User Management](#) (page 294)

This section describes the options that can be made available to PureMessage end users, the people within your organization who are the senders and recipients of email that is processed by PureMessage.

[Policy Rules](#) (page 251)

[Digest Configuration](#) (page 287)

[Operators](#) (page 91)

[Wildcard Usage](#) (page 117)

Related tasks

[Creating Lists or Maps](#) (page 114)

[Editing Lists](#) (page 120)

[Testing Lists or Maps](#) (page 121)

Related information

[maps.conf](#)

CDB Lists and Maps

PureMessage supports lists and maps in CDB format. This on-disk format is useful for large lists and maps (that is, 5,000 entries or more), where the default plain text format can cause excessive memory consumption and latency.

To convert lists or maps to CDB format:

1. Edit the list or map configuration file manually. Replace `'source = file:'` with `'source = cdbfile:'`. The `match_type` for CDB lists and maps must be either `'is'` or `'mail-parts'`. For example, in `pmx/etc/lists.conf`:

```
<list internal-hosts>
  name = "Internal hosts"
  description = "Relay hosts regarded as internal"
  precious = yes
  source = cdbfile:internal-hosts
  match_type = is
</list>
```


In `pmx/etc/maps.conf`:

```
<map notifications>
  name = "Notifications address map"
  description = "Notifications about messages processed
  by PureMessage..."
  source = cdbfile:notifications
  match_type = mail-parts
</map>
```

Note

The 'mail-parts' match type is a simple substring match that does not accept regular expressions or wildcards. For example:

```
someuser@example.com
someuser
@example.com
```

Without an @ sign, the list entry matches a username (ignoring the domain name).

2. Check to make sure that the list or map does not contain syntax that is not supported by the `match_type` used.
3. Use `pmx-makemap` to compile the CDB files:

```
pmx-makemap --all
```

When lists and maps from a central server are synchronized to edge servers in multi-server deployments (using either `pmx-profile` or publications), a scheduled job must be created, or an existing job modified, to compile the CDB lists locally with `pmx-makemap` after synchronization. For example, the `resource-sync` scheduled job can be modified as follows:

```
pmx-profile sync-from-db --clean; pmx-makemap --all
```

The `resource-sync` job will only synchronize files if they have changed on the central server, and `pmx-makemap` will only compile when the local file has changed.

Related concepts

[Managing Publications](#) (page 184)

Related tasks

[Managing Scheduled Jobs](#) (page 172)

Related information

[pmx-makemap](#)

[pmx-profile](#)

Policy Repository

The Policy Repository stores [Sieve](#) rules that can be added to the PureMessage policy script. By default, the repository contains a number of general-purpose snippets that can be copied and pasted to the PureMessage policy. Conversely, snippets can be added to the Policy Repository by copying and pasting rules from the PureMessage policy to the Clipboard or by adding Sieve files from the command line. The Policy Repository can be used to store complete Sieve scripts as well as snippets. For more about working with snippets, see "Managing the Repository" in the *Manager Reference*.

Related concepts

[Managing the Repository](#) (page 106)

Related information

[pmx-policy](#)

Populating Lists and Maps via LDAP

Lightweight Directory Access Protocol (LDAP) is an open-standard protocol for accessing online directory services. Directory services are structured repositories of information on people and resources within an organization (for example, a list of names and email addresses). LDAP defines a protocol for updating and searching these directory services running over TCP/IP. For information on configuring an LDAP directory service see the following resources:

- LDAP RFC 1777, Request for Comments documentation.
- OpenLDAP.org, the open source implementation of the Lightweight Directory Access Protocol.

Use the `pmx-ldap-sync` program to synchronize the existing LDAP directory service to a PureMessage list (for example, a whitelist or blacklist) or map. Depending on options specified on the command line, the `pmx-ldap-sync` program creates either a flat file or a Berkeley database from an LDAP directory service. Use Perl regular expressions to evaluate list content and filter it based on specific criteria. The `pmx-ldap-sync` program can be run as a scheduled job from the Manager; see "Managing Scheduled Jobs" in the *Manager Reference* for more information.

Important

Sophos highly recommends that only administrators with advanced LDAP configuration and query experience use the `pmx-ldap-sync` program. Administrators must also be familiar with Perl and regular expressions. Accessing LDAP directory services and writing LDAP queries is not included in the Sophos PureMessage support agreement.

Related tasks

[Managing Scheduled Jobs](#) (page 172)

Related information

[LDAP RFC 1777](#)

[OpenLDAP.org](#)

[pmx-ldap-sync](#)

Customizing Policies

PureMessage policy configuration can be easily tailored to meet your corporate needs. When customizing policy rules, be aware of the following considerations:

- Backup: Before editing a policy, it is strongly advised that you make a backup of the current policy.
- Committing Changes: When using the PureMessage Manager to edit policies, you must "commit" changes before testing or enabling the policy.
- Rule Order: Rules are processed in the order they are configured. When a message triggers the "stop" action in a rule, the message is no longer processed. Thus, the rule order and the placement of "stop" actions impact the efficiency of the policy.
- Testing: It is advisable to test policies before enabling them.

The file `/opt/pmx6/bin/pmx-policy` provides an interface to the PureMessage policy engine.

Related concepts

[Backing Up and Reverting Policies](#) (page 270)

[Editing the Policy](#) (page 80)

[Testing Policies](#) (page 270)

Related tasks

[Configuring and Distributing Policy Settings](#) (page 187)

Related information

[pmx-policy](#)

Example: Check Messages for Offensive Words

Rules can be configured to implement policies that reduce liability from inappropriate communications. The following rule checks messages originating from internal sources against a defined list of offensive words, and it quarantines any messages found to contain those words. (Note that this example assumes that you are using the default PureMessage Policy configuration.)

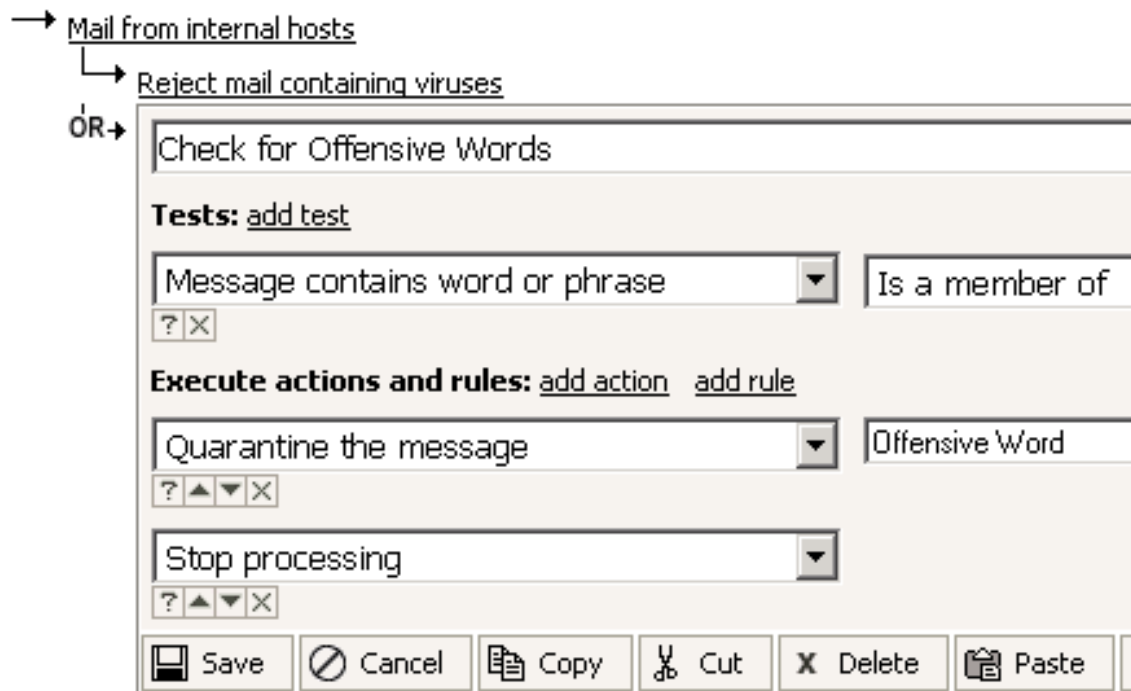
The Offensive Words list is pre-configured with several hundred words and phrases that are generally considered to be unacceptable in corporate communications. Before creating a rule that uses the Offensive Words list, review the contents of the list, adding words and phrases specific to your organization and removing words and phrases that are not applicable.

The following test uses the **Message contains word or phrase** test, which is part of the PureMessage Policy bundle. If you do not have a license for the Policy bundle, use the **Message has offensive content** test, which also uses the Offensive Words list, and which is included with the Anti-Spam license.

To create a "check messages for offensive words" rule using the PureMessage Manager:

1. Click the **Create** link beside the **Backups** text on the sidebar of the **Policy** tab.

A backup of the current policy is created; a backup entry with the current date and time is displayed in the **Backups** section of the sidebar.
2. PureMessage includes a pre-configured list of offensive words and phrases. To view the contents of this list, click **Offensive Words** in the **Lists** section of the **Policy** tab sidebar.
3. Click **Policy Rules** on the sidebar of the **Policy** tab to display the current policy.
4. Click the **Mail from internal hosts** rule.
5. Click the **add rule** link (beside **Execute actions and rules**). This creates a new rule at the bottom of the **Internal Hosts** section of the PureMessage Policy.
 - a) Configure the Test:
 - I) Change the (**New Rule**) text to *Check for Offensive Words*.
 - II) From the **Tests** drop-down list, select **Message contains word or phrase**. Select **Is a member of** as the operator and **Offensive Words** from the available lists.
 - b) Configure the Action:
 - I) Click **add action**. This creates the action configuration template.
 - II) In the new **Execute actions and rules** drop-down list, select **Quarantine the message**.
 - III) In the text box to the right (**Quarantine Reason**), enter *Offensive Word*.
 - IV) In the second rules drop-down list, select **Stop processing**.
 - c) Click **Save**.
 - d) Click the **Commit** link to update the live policy script. PureMessage displays a message advising that the militer is running with a stale configuration. Do not restart the militer.



6. Test New Policy: Because the militer has not been restarted, it is still using the original policy. Therefore, the new policy can be tested without making it "live".
 - a) Click **Test Current Policy** on the sidebar of the **Policy** tab.
 - b) Enter `adult site` in the message source text box. (The default relay type is **Internal**, which is as desired.)
 - c) Click **Test**. The test runs and the test results are displayed. Note that the **Delivery Actions** for the test message is **quarantine: Offensive Word**.
7. If satisfied with the new policy, click **Restart now** to restart the militer and make the new policy live. To restore the original policy, click the backup link, and select **OK**.

Policy Script

To create a check messages for offensive words rule by editing the policy script:

```
if pmx_virus {
  # attr NAME=Allow unscannable messages to pass through
  if pmx_virus_cantscan {
    keep;
    stop;
  }
  reject "One or more viruses were detected in the message.";
  stop;
}
# attr NAME=Check for Offensive Words
elsif pmx_phrase :memberof ["offensive-words"] {
  pmx_quarantine "Offensive Word";
  stop;
}
```

See the Policy Script Tutorial for more information about modifying the policy script from the command line.

Related concepts

[The Default PureMessage Policy](#) (page 243)

[Lists](#) (page 252)

[Policy Tab](#) (page 78)

[About PureMessage Default Lists](#) (page 118)

[Testing Policies](#) (page 270)

[Policy Script Tutorial](#) (page 314)

This tutorial describes the syntax used in the policy script, analyzes the default PureMessage policy script, and shows examples of common policy script modifications.

Related tasks

[Creating and Restoring Policy Backups](#) (page 79)

Example: Add Corporate Information to Outbound Messages

It is common for corporate email messages to carry standard tag lines that are appended to outgoing messages. The following rule automatically adds specified information to messages sent to external addresses, but not to messages exchanged within a company. (Note that this example assumes you are using the default PureMessage Policy configuration.)

To add corporate information to outbound messages using the PureMessage Manager:

1. First, create a list containing the email addresses for all internal mail system users:
 - a) click **New** beside **Lists** on the **Policy** tab sidebar. The **Add List/Map** page is displayed.
 - b) From the **Type** drop-down list, select **List**.
 - c) In the **ID** text box, enter `Local`.
 - d) In the **Name** text box, enter `Local Users`.
 - e) In the **Description** text box, enter `Company email addresses`.
 - f) From the **Match Type** list, select **Exact**.
 - g) Click **Save**. You are prompted to add items to the list. Click **here** to display the **Edit List** page.
 - h) Enter the desired email addresses in the **Add Items** text box, be sure that each entry appears on a separate line.
 - i) Click **Add**. The email addresses are included under **List Items**.
2. Next, create a policy that appends a tag line inviting feedback from all external mail recipients:
 - a) Click **Create** beside **Backups** on the **Policy** tab sidebar. A backup of the current policy is created; a backup entry with the current date and time are displayed in the **Backups** section of the sidebar.
 - b) On the **Policy** tab, click the **Mail from internal hosts** rule.
 - c) Click **add rule** (beside **Execute actions and rules**). This creates a new rule at the bottom of the **Internal Hosts** section of the PureMessage Policy.
 - I) Configure the Test:
 - i) Change the **(New Rule)** text to `Add message inviting feedback`.
 - ii) From the **Tests** drop-down list, select **Recipient's address**. Select **Is a not member of** as the operator and **Local Users** from the available lists.
 - II) Configure the Action:
 - i) Click **add action**. This creates the action configuration template.
 - ii) In the new **Execute actions and rules** drop-down list, select **Add Banner**.
 - iii) Click the **Arguments** button to the right of the actions drop-down list.
 - iv) Select the **Append banner to message body** check box.
 - v) In the **Data_type** text box, select **Filename**.

- vi) Using a text editor, create and save a file containing the following:

```
At Acme, we are always looking for ways to serve you better.
We encourage you to send any questions or comments to
"feedback@acme.com" or call our customer service department at
1-800-000-0000.
```

- vii) In the **File or string** text box, enter the path to the file created in the previous step. Click **OK**.

Add banner

☒ Append banner to message body

☐ Add banner to specified header

☐ Ignore these content-types

☐ Enclose banner in HTML <PRE> tags

Data_type

File or string

Cancel Ok Revert

- viii) In the second rules drop-down list, select **Stop processing**.

- III) Click **Save**

→ Mail from internal hosts

↳ Reject mail containing viruses

OR →

Add message inviting feedback

Tests: add test

Recipient's address Is not a member of

Execute actions and rules: add action add rule

Add banner Arguments...

Stop processing

- IV) Click the **Commit** link to update the live policy script. PureMessage displays a message advising that the milter is running with a stale configuration. Do not restart the milter (so you can test the changes without making them live).
- d) Test New Policy: Because the milter has not been restarted, it is still using the original policy. Therefore, the new policy can be tested without making it "live".
 - I) Click **Test Current Policy** link on the **Policy** tab sidebar.
 - II) In the **Envelope from** text box, type a sender's email address.
 - III) In the **Envelope to** text box, type an email address that is not included in your **Local Users** list.
 - IV) Click **Test**. The test runs, and the test results are displayed. In the **Details** window, notice there are **POLICY RULE HIT** entries for **Mail from internal hosts** and **Add message inviting feedback**, and that there is a **POLICY ACTION** entry indicating that a banner has been added.
- e) If satisfied with the new policy, click **Restart now** to restart the milter and make the new policy live. To restore the original policy, click the backup link, and select **OK**.

Policy Script

To add corporate information to outbound messages by editing the policy script:

```
if pmx_virus {
    # attr NAME=Allow unscannable messages to pass through
    if pmx_virus_cantscan {
        keep;
        stop;
    }
    reject "One or more viruses were detected in the message.";
    stop;
}
# attr NAME=Add message inviting feedback
elsif not address :all :memberof :comparator "i;ascii-casemap" ["to",
                                                                "cc",
                                                                "bcc"]
                                                                ["Local"]
{
    pmx_add_banner :body :file "/opt/pmx6/home/banner.txt";
    stop;
}
```

See the Policy Script Tutorial for more information about modifying the policy script from the command line.

Related concepts

[The Default PureMessage Policy](#) (page 243)

[Policy Tab](#) (page 78)

[Testing Policies](#) (page 270)

[Policy Script Tutorial](#) (page 314)

This tutorial describes the syntax used in the policy script, analyzes the default PureMessage policy script, and shows examples of common policy script modifications.

Related tasks

[Creating and Restoring Policy Backups](#) (page 79)

Example: Archive Messages from a Group of Users

For some types of email correspondence, it is desirable to maintain an archive of all messages that originate from a specific address or group of addresses within a company. The following rule creates copies of all messages sent by a member of a designated list and sends them to the quarantine to be archived. (Note that this example assumes that you are using the default PureMessage Policy configuration.)

To archive messages from a group of users using the PureMessage Manager:

1. First, create a list containing the email addresses for the group of users whose messages you want to archive:
 - a) Click **New** beside **Lists** on the **Policy** tab sidebar. The **Add List/Map** page is displayed.
 - b) From the **Type** drop-down list, select **List**.
 - c) In the **ID** text box, enter `lg1`.
 - d) In the **Name** text box, enter `Legal`.
 - e) In the **Description** text box, enter Email addresses of Legal Department employees.
 - f) From the **Match Type** list, select **Exact**.
 - g) Click **Save**. You are prompted to add items to the list. Click [here](#) to display the **Edit List** page.
 - h) Enter the desired email addresses in the **Add Items** text box, making sure that each entry appears on a separate line.
 - i) Click **Add**. The email addresses are included under **List Items**.
2. Next, create a policy that will archive messages from the group of users specified in the **Legal** list:
 - a) Click **Policy Rules** on the **Policy** tab sidebar to display the current policy.
 - b) Click **Create** beside **Backups** on the **Policy** tab sidebar. A backup of the current policy is created; a backup entry with the current date and time is displayed in the **Backups** section of the sidebar.
 - c) On the **Policy** tab, click the **Mail from internal hosts** rule.
 This creates a new rule at the bottom of the **Internal Hosts** section of the PureMessage Policy.
 - I) Configure the Test:
 - i) Click **add rule** (beside **Execute actions and rules**). This creates a new rule at the bottom of the **Internal Hosts** section of the PureMessage Policy.
 - ii) Change the **(New Rule)** text to **Archive Messages from Legal**.
 - iii) From the **Tests** drop-down list, select **Envelope from**. Select **Is a member of** as the operator and **Legal** from the available lists.
 - II) Configure the Action:
 - i) Click **add action**. This creates the action configuration template.
 - ii) In the new **Execute actions and rules** drop-down list, select **Copy the message to quarantine**.
 - iii) In the text box on the right (**Quarantine Reason**), enter `Legal Archive`.
 - iv) In the second rules drop-down list, select **Stop processing**.
 - III) Click **Save**.

→ Mail from internal hosts

→ Reject mail containing viruses

OR →

Archive messages from Legal

Tests: [add test](#)

Envelope from Is a member of

[?](#) [X](#)

Execute actions and rules: [add action](#) [add rule](#)

Copy the message to quarantine Legal Archive

[?](#) [▲](#) [▼](#) [X](#)

Stop processing

[?](#) [▲](#) [▼](#) [X](#)

- IV) Click the **Commit** link to update the live policy script. PureMessage displays a message advising that the mitler is running with a stale configuration. Do not restart the mitler (so you can test the changes without making them live).
- d) Test New Policy: Because the mitler has not been restarted, it is still using the original policy. Therefore, the new policy can be tested without making it "live".
- I) Click **Test Current Policy** on the **Policy** tab sidebar.
- II) In the **Envelope from** text box, type one of the email addresses added to the **Legal** list.
- III) In the message source text box, replace the default text with the text for the test message.
- IV) Click **Test**. The test runs and the test results are displayed. Notice there are two resulting delivery actions, **copy to quarantine** and **keep**. In the **Details** window, notice that there is a **POLICY RULE HIT** entry for "Archive messages from legal" and a **POLICY ACTION** entry indicating that a copy of the message has been filed in the quarantine.

Note

By default, messages remain in the quarantine for seven days before PureMessage moves them to the `/opt/pmx6/home/archive` directory. Manually delete unwanted messages from this archive. While managing quarantined messages, however, be careful not to prematurely delete messages copied to the quarantine for the purpose of archiving.

- e) If satisfied with the new policy, click **Restart now** to restart the mitler and make the new policy live. To restore the original policy, click the backup link, and select **OK**.

Policy Script

To archive messages from a group of users by manually editing the policy script:

```
if pmx_virus {
    # attr NAME=Allow unscannable messages to pass through
    if pmx_virus_cantscan {
        keep;
        stop;
    }
    reject "One or more viruses were detected in the message.";
    stop;
}
# attr NAME=Archive messages from Legal
elseif envelope :comparator "i;ascii-casemap" :all :memberof ["from"]
                                                                ["Legal"]
{
    pmx_file "Legal Archive";
    stop;
}
```

See the Policy Script Tutorial for more information about modifying the policy script from the command line.

Related concepts

[The Default PureMessage Policy](#) (page 243)

[Policy Tab](#) (page 78)

[Managing the Quarantine](#) (page 129)

[Testing Policies](#) (page 270)

[Policy Script Tutorial](#) (page 314)

This tutorial describes the syntax used in the policy script, analyzes the default PureMessage policy script, and shows examples of common policy script modifications.

Related tasks

[Creating and Restoring Policy Backups](#) (page 79)

Example: Quarantine Messages from Fake Senders

Some spammers falsify email addresses so that the message appears to originate from a sender within the recipient's own domain. For example, the recipient works for company "XYZ", and his own corporate email address is `john@xyz.com`. A spammer might then pose as `frank@xyz.com` in an attempt to evade detection.

Guard against this tactic by adding a rule to the "Mail from external hosts" section of the policy that filters incoming mail from senders pretending to be a member of the same domain. The rule tests external messages for the presence of the domain in the `Envelope from` and `Sender` parts. Any messages containing the specified domain are quarantined.

Note

This test is based on the premise that legitimate members of the domain require some form of authentication to access their email accounts externally. Therefore, it is assumed that addresses with the company's domain that originate from outside of the network are fake.

To Quarantine Messages from Fake Senders using the PureMessage Manager:

1. Click **Create** beside **Backups** on the **Policy** tab sidebar. A backup of the current policy is created; a backup entry with the current date and time are displayed in the **Backups** section of the sidebar.

2. Click the **Policy Rules** on the **Policy** tab sidebar to display the current policy.
3. Click the **Mail from external hosts** rule.
4. Click **add rule** (beside **Execute actions and rules**). A new rule is created.
 - a) Configure the Test:
 - I) Change the (**New Rule**) text to **Check for fake senders**.
 - II) From the **Tests** drop-down list, select **Sender's address**.
 - III) From the second drop-down list, select **Matches**.
 - IV) In the adjacent text box, enter ****@xyz.com**.
 - V) Click **add test**.
 - VI) From the drop-down list, select **Envelope from**.
 - VII) From the second drop-down list, select **Matches**.
 - VIII) In the adjacent text box, enter ****@xyz.com**.
 - IX) From the criteria drop-down list, select **If ANY criteria are met**.
 - b) Configure the Action:
 - I) Click **add action**. This creates the action configuration template.
 - II) In the **Execute actions and rules** drop-down list, select **Quarantine the message**.
 - III) In the text box on the right (**Quarantine Reason**), enter **Fake Sender**.
 - IV) Click **add action**.
 - V) In the second **rules** drop-down list, select **Stop processing**.
 - VI) Click **Save**.
 - c) Change the Rule Order:
 - I) Click **Cut**. A message is displayed at the top of the page indicating that the **Check for fake senders** rule has been cut.
 - II) Click to select the rule **Clean mail containing viruses**.
 - III) Click **Paste**. The **Check for fake senders** rule is now displayed in its new position beneath the **Check mail containing viruses** rule.

Mail from external hosts

Clean mail containing viruses

Check for fake senders

Tests: [add test](#)

Sender's address ▼ Matches ▼

? ▲ ▼ X

Envelope from ▼ Matches ▼

? ▲ ▼ X

If ANY criteria are met ▼

Execute actions and rules: [add action](#) [add rule](#)

Quarantine the message ▼ Fake Sender

? ▲ ▼ X

Stop processing ▼

? ▲ ▼ X

Save Cancel Copy Cut Delete Paste Add

- d) Click the **Commit** link to update the live policy script. PureMessage displays a message advising that the militer is running with a stale configuration. *Do not* restart the militer.
5. Test New Policy: Because the militer has not been restarted, it is still using the original policy. Therefore, the new policy can be tested without making it "live".
 - a) Click **Test Current Policy** on the **Policy** tab sidebar. The **Test Current Policy** page is displayed.
 - b) From the **Select Relay Type** drop-down list, select **External**.
 - c) In the **Envelope From** text box, enter: frank@xyz.com.
 - d) *Do not* edit the default text displayed in the **message source** text box.
 - e) Click **Test**. The test runs and the results are displayed. Note that the **Delivery Action** for the test message is "quarantine: Fake_Sender". Scroll down in the **Details** list box to view the test results.
6. If satisfied with the new policy, click **Restart now** to restart the militer and make the new policy live. To restore the original policy, click the backup link, and select **OK**.

Policy Script

To Quarantine Messages from Fake Senders by manually editing the policy script:

```
# attr NAME=Mail from external hosts
else {
    pmx_add_header "X-PMX-Version" "%PMX_VERSION%";
    pmx_mark "Size" "%MESSAGE_SIZE%";
    # attr NAME=Clean mail containing viruses
    if pmx_virus {
        pmx_file "Virus";
        pmx_virus_clean "cantclean.tmpl";
        pmx_replace_header "Subject" "[PMX:VIRUS] %SUBJECT%";
        pmx_replace_header "X-PerlMx-Virus-Detected" "%VIRUS_IDS%";
    }
    # attr NAME=Check for fake senders
    if anyof(address :all :matches :comparator "i;ascii-casemap" ["from"]

["**xyz.com"],
    envelope :comparator "i;ascii-casemap" :all :matches ["from"]
                                                                ["**xyz.com"] )
    {
        pmx_quarantine "Fake Sender";
        stop;
    }
}
```

See the Policy Script Tutorial for more information about modifying the policy script from the command line.

Example: Run Per-Recipient Tests

Create a rule that causes PureMessage to perform different actions, depending on the recipient of the message. When such a rule is applied, PureMessage splits a message addressed to multiple recipients into copies and operates on the copies independently. This makes it possible to specify different actions for the individual recipients of messages addressed to multiple recipients.

Per-recipient rules are created using Envelope from and Envelope to tests. In this example, you will create a rule that adds a banner to messages addressed to the customer service representatives of company "XYZ".

To Run Per-Recipient Tests using the PureMessage Manager:

1. Click **Create** beside **Backups** on the **Policy** tab sidebar. A backup of the current policy is created, and the current date and time are displayed in the **Backups** section of the sidebar.
2. Click **Policy Rules** on the **Policy** tab sidebar to display the current policy.
3. Click the **Mail from external hosts** rule.
4. Click **add rule** (beside **Execute actions and rules**). This creates a new rule.
 - a) Configure the Test:
 - I) Change the **(New Rule)** text to Add banner for selected recipients only.
 - II) From the **Tests** drop-down list, select **Envelope to**.
 - III) From the second drop-down list, select **Is**.
 - IV) In the adjacent text box, enter:

```
"kurt@service.xyz.com", "kris@service.xyz.com",
"dave@service.xyz.com"
```

- b) Configure the Action:
 - I) In the **Execute actions and rules** drop-down list, select **Add banner**.

- II) Click the **Arguments** button to the right of the drop-down list.
 - III) Select the **Append banner to message body** check box.
 - IV) Select **Verbatim** from the **Data_type** drop-down list.
 - V) In the **File or string** text box, enter: ATTENTION
 - VI) Click **OK**.
 - VII) Click **Save**.
- c) Change the Rule Order:
- I) Click **Cut**. A message is displayed at the top of the page indicating that the "Add banner for selected recipients only" rule has been cut.
 - II) Click the rule **Clean mail containing viruses**.
 - III) Click **Paste**. The "Add banner for selected recipients only" rule is displayed in its new position beneath the "Clean mail containing viruses" rule.

Mail from external hosts

└─ Clean mail containing viruses

└─ Add banner for selected recipients only

Tests: [add test](#)

Envelope to Is

[?](#) [X](#)

Execute actions and rules: [add action](#) [add rule](#)

Add banner Arguments...

[?](#) [▲](#) [▼](#) [X](#)

Accept the message

[?](#) [▲](#) [▼](#) [X](#)

Save Cancel Copy Cut Delete Paste Add

- d) Click the **Commit** link to update the live policy script. PureMessage displays a message advising that the milner is running with a stale configuration. *Do not* restart the milner.
5. Test New Policy: The milner uses the original policy until the service is restarted. However, the new policy can be tested without making it "live".
- a) Click **Test Current Policy** on the **Policy** tab sidebar. The **Test Current Policy** page is displayed.
 - b) From the **Select Relay Type** drop-down list, select **External**.
 - c) In the **Envelope From** text box, enter:

customer@example.com

- d) In the **Envelope To** text box, enter:

kurt@service.xyz.com, jane@sales.xyz.com

- e) Accept the sample text that is displayed by default in the **message source** text box.

- f) Click **Test**. The test runs and the results are displayed. Note that the **Delivery Action** for each of the two resulting messages is "keep".
 - g) Click the number of the first message in the **Resulting Message** column. Notice that the banner text (**ATTENTION**) appears beneath the message body text.
 - h) In the **Message Preview** window, click the **Quarantine Info** tab. Notice that the **Envelope From** and **Envelope To** details are displayed in the **Quarantine Info** table.
 - i) Click the number of the second message in the **Resulting Message** column to view details of the message for which a banner was not added.
6. If satisfied with the new policy, click **Restart now** to restart the filter and make the new policy live. To restore the original policy, click the backup link, and select **OK**.

Policy Script

To run per-recipient tests by manually editing the policy script:

```
# attr NAME=Mail from external hosts
else {
    pmx_add_header "X-PMX-Version" "%%PMX_VERSION%%";
    pmx_mark "Size" "%%MESSAGE_SIZE%%";
    # attr NAME=Clean mail containing viruses
    if pmx_virus {
        pmx_file "Virus";
        pmx_virus_clean "cantclean.tmpl";
        pmx_replace_header "Subject" "[PMX:VIRUS] %%SUBJECT%%";
        pmx_replace_header "X-PerlMx-Virus-Detected" "%%VIRUS_IDS%%";
    }
    # attr NAME=Add banner for selected recipients only
    if envelope :comparator "i;ascii-casemap" :all :is ["to"]
["kurt@service.xyz.com",

"kris@service.xyz.com",

"dave@service.xyz.com"]
    {
        pmx_add_banner :body :data "ATTENTION";
        keep;
    }
}
```

See the Policy Script Tutorial for more information about modifying the policy script from the command line.

Related concepts

[Policy Tab](#) (page 78)

[Testing Policies](#) (page 270)

[Policy Script Tutorial](#) (page 314)

This tutorial describes the syntax used in the policy script, analyzes the default PureMessage policy script, and shows examples of common policy script modifications.

Related tasks

[Creating and Restoring Policy Backups](#) (page 79)

Testing Policies

After changing the policy configuration, it is advisable to test the policy before making it live. Tests can be run in the PureMessage Manager, or using the `pmx-test` command-line program. Note the following variations between these methods:

Testing via the PureMessage Manager:

- Policy changes must be "committed" before running tests.
- The militer does not require a "restart" before running tests; therefore, policy changes can be tested without making the policy "live".
- A record of the test messages is not stored in the message log; therefore, report statistics are not affected.
- Test messages are not added to the quarantine.

Testing via the Command Line (`pmx-test`):

- The militer must be restarted before running tests.
- A record of the test messages is stored in the message log; therefore, report statistics may be affected.
- Test messages are added to the quarantine.

Related tasks

[Testing the Current Policy](#) (page 112)

Related information

[pmx-test](#)

Backing Up and Reverting Policies

Before altering the policy configuration, Sophos recommends making a backup of the current policy that you can revert to if necessary. In addition, policy backups allow you to create a number of different policies, and then experiment to determine which produces the best results.

The PureMessage Manager provides an interface for backing up, restoring and comparing policies. You can also create a backup from the command line by making a copy of the `/opt/pmx6/etc/policy.siv` file.

Reverting to the Default Policy Script

To recover the default `policy.siv` script, run the following command as the PureMessage user ("pmx6" by default).

```
pmx-policy integ --regen
```

This command saves the current `policy.siv` file to a backup file named `.policy.siv.bak<timestamp>`, and regenerates the default `policy.siv` file. The militer must be restarted using the `pmx-restart` command to activate the new version of `policy.siv`. The backup file is displayed in the **Backups** section of the PureMessage Manager's **Policy** tab.

Related tasks

[Creating and Restoring Policy Backups](#) (page 79)

Applying Policies to the Contents of the Quarantine

After changing the policy configuration, process the contents of the quarantine using the new configuration. This can only be done using the command-line `pmx-policy` program, not the PureMessage Manager.

The relevant commands and arguments for the `pmx-policy` program are as follows:

- `qinject` : "Injects" the messages in the quarantine into the policy engine.
- `--dry-run` : The "test" switch; the original messages are not delivered, discarded, or deleted from the quarantine.
- `--delete` : Deletes the original message from the quarantine.
- `--where` : Specifies which messages in the quarantine should be processed.

For example, to test the spam messages in the quarantine, use the following command:

```
pmx-policy qinject --dry-run --where "any m_reason == 'spam'"
```

To do a "live" run of the policy against all messages in the quarantine with a content type that starts with `text/`, enter:

```
pmx-policy qinject --delete --where "c_content_type like 'text/%'"
```

For a complete description of the syntax for the `--where` switch, enter `perldoc PureMessage::MessageStore` on the command line.

Related concepts

[Quarantine Administration](#) (page 280)

This section describes management of the PureMessage quarantine, a temporary holding place for messages that are deemed potentially problematic by the PureMessage policy. Quarantined messages can then be reviewed, and released or deleted.

Related information

[pmx-policy](#)

3.5.2 Spam Detection

PureMessage identifies spam by analyzing messages according to a set of anti-spam rules. Each rule has a test and a corresponding "weight". For each rule that matches the message, the weight is added to the message's total spam score. After all rules are applied, the spam score is converted to a percentage. The PureMessage policy performs actions (such as quarantining a message) based on the percentage that expresses the message's total spam score.

The PureMessage applications and configuration files used to configure spam detection from the command line are:

- `/opt/pmx6/bin/pmx-spam`: An interface to the PureMessage anti-spam component.

Anti-Spam Policy Related Configuration Files

PureMessage spam detection uses a number of 'feature groups'. Each feature group implements a different method of message analysis. One or more feature groups can be enabled at the same time. Feature groups are enabled via the configuration files stored in the `/opt/pmx6/etc/spam.d/compile.d` directory.

The `spam.conf` configuration file sets general message-scanning parameters for all feature groups. These general configuration options are combined with the feature-group-specific options in the other configuration files.

After altering anti-spam configuration, enabling or disabling a feature group, or adding or modifying rules, you must re-start the PureMessage milter (using the command `pmx-milter restart`) in order for the changes to take effect.

- `/opt/pmx6/etc/spam.conf` : Contains general anti-spam configuration items that apply regardless of which feature groups (methods of analysis) are enabled.
- `/opt/pmx6/etc/spam.d/compile.d/destination.conf` : Enables the Known Spam Destination feature group.
- `/opt/pmx6/etc/spam.d/compile.d/heuristic.conf` : Enables the Heuristic Analysis feature group.
- `/opt/pmx6/etc/spam.d/compile.d/sender.conf` : Enables the Sender Reputation feature group.
- `/opt/pmx6/etc/spam.d/net.conf` : Sets the parameters for DNS checks used by the Sender Reputation feature group.
- `/opt/pmx6/etc/spam.d/dnsbl.conf` : Sets the parameters for DNSBL (DNS black list) checks used by the Sender Reputation feature group (the black lists in this group are disabled by default).
- `/opt/pmx6/etc/spam.d/compile.d/site.conf` : Enables the Site Features feature group.
- `/opt/pmx6/etc/spam.d/sxl.conf` : Enables real-time, DNS-based queries for Sophos anti-spam data.

Rules and Custom Rules Configuration Files

PureMessage is distributed with a set of pre-configured anti-spam rules. These rules are regularly updated as part of the PureMessage Anti-Spam heuristic update. Only the weight and probability delta can be altered for default rules; these alterations are done using the `pmx-spam` program (see the `pmx-spam` man page for more information).

Custom rules are stored in the `re.rules` file, located in the `etc/spam.d` directory located beneath the default PureMessage installation directory. Custom rule files are never updated as part of the PureMessage Anti-Spam heuristic update.

When rules are applied to messages, both default and custom rules are used.

Rule status (enabled or disabled), weights and probabilities are stored in a database, rather than in the rule definition files. To adjust rule weights, use the `pmx-spam` program.

- `/opt/pmx6/etc/spam.d/re.rules` : Stores custom rules.
- `/opt/pmx6/etc/spam.d/compile.d/compiler.conf` : Sets a threshold number below which a spam identification rule is not used.

Related concepts

[Policy Overview](#) (page 242)

Related information

[pmx-spam](#)

[spam.conf](#)

[destination.conf](#)

[heuristic.conf](#)

[sender.conf](#)

[net.conf](#)

[dnsbl.conf](#)

[site.conf](#)
[re.rules](#)
[compiler.conf](#)
[sxl.conf](#)

About Anti-Spam Rules

The PureMessage anti-spam rules are displayed on the Anti-Spam Rules page of the PureMessage Manager. Default rules and scores are stored in the `etc/data/antispam` directory (in binary files), beneath the PureMessage installation directory. Site-specific custom rules are stored in the `etc/spam.d` directory.

Anti-spam rules consist of a test definition and a "weight". If the test matches the message, the corresponding weight is added to the message's total spam score. Generally, multiple rules must be triggered by a message in order to result in a spam score high enough for an action to be taken by the policy filter.

Note

Anti-spam rules cannot be used to check message attachments for viruses. This must be done via virus detection.

The PureMessage development team constantly analyzes emerging spam techniques and updates the PureMessage anti-spam rule set accordingly. See [Check For Updates](#) in the PureMessage Manager Reference for information about installing updates.

Help Sophos in its continuous efforts to improve the accuracy of spam [heuristics](#) by forwarding misidentified items as attachments to:

- Missed Spam: is-spam@labs.sophos.com
- Not Spam: not-spam@labs.sophos.com

Related concepts

[Managing Anti-Spam Rules](#) (page 124)

[The Default PureMessage Policy](#) (page 243)

[Viruses](#) (page 278)

[Available Updates](#) (page 189)

Test Types

PureMessage uses a variety of methods to detect spam. These methods are embedded in the test definition of anti-spam rules. PureMessage detection methods are included in one of the following feature groups.

- **Spam Signatures Analysis:** Signatures are created using spam data compiled by SophosLabs. There are signatures for each of the various email message parts, including the message body, paragraphs within the message body, HTML, images, and attachments. These are all tested against the contents of the messages that PureMessage processes. Signatures can be used to detect spam characteristics, even during spam campaigns in which some aspects of the messages are still evolving.
- **Known Spam Destinations:** PureMessage includes a database of URLs associated with spam messages. This database is distributed with the PureMessage Anti-Spam heuristic update. URI tests determine whether messages contain URLs that are included in this database.
- **Sender Reputation:** By default, PureMessage performs two types of DNS checks: reverse DNS look-ups and queries via Sophos's own [SXL](#) infrastructure. Because IP classification is handled as

part of the SXL queries, the third-party DNS black lists are disabled by default. Before enabling any of the disabled DNSBL rules, be sure that you have the associated DNSBL licenses, if applicable.

- **Heuristic Analysis:** Specified message components, such as the subject line or the message body, are analyzed by a regular expression. For example, a regular expression test can check for the occurrence of a specific word or phrase in the body of an email message. On the Anti-Spam Rules page in the PureMessage Manager, the test definition component of a regular expression test is prefixed by the word `Test`.
- **Site Features:** PureMessage uses internal programmatic functions to test for various message characteristics. For example, multiple similar recipient addresses (`johna@domain.com`, `johnb@domain.com`, `johnc@domain.com`) often indicate a spam message. On the Anti-Spam Rules page in the PureMessage Manager, the test definition component of a message evaluation test is prefixed by the word `Eval`.

PureMessage also uses "meta" tests to check the result of two or more tests. For example, a meta test might be configured to be true if two other rules are also true. You cannot create custom meta tests; however, you can alter the score of existing tests. On the Anti-Spam Rules page in the PureMessage Manager, the test definition component of a meta test is prefixed by the word `Meta`.

Regular expression tests, message evaluation tests, meta tests and URI tests are enabled by default, as are DNS checks. tests generated from adaptive classification training must be manually configured and enabled.

Related concepts

[Managing Anti-Spam Rules](#) (page 124)

Related information

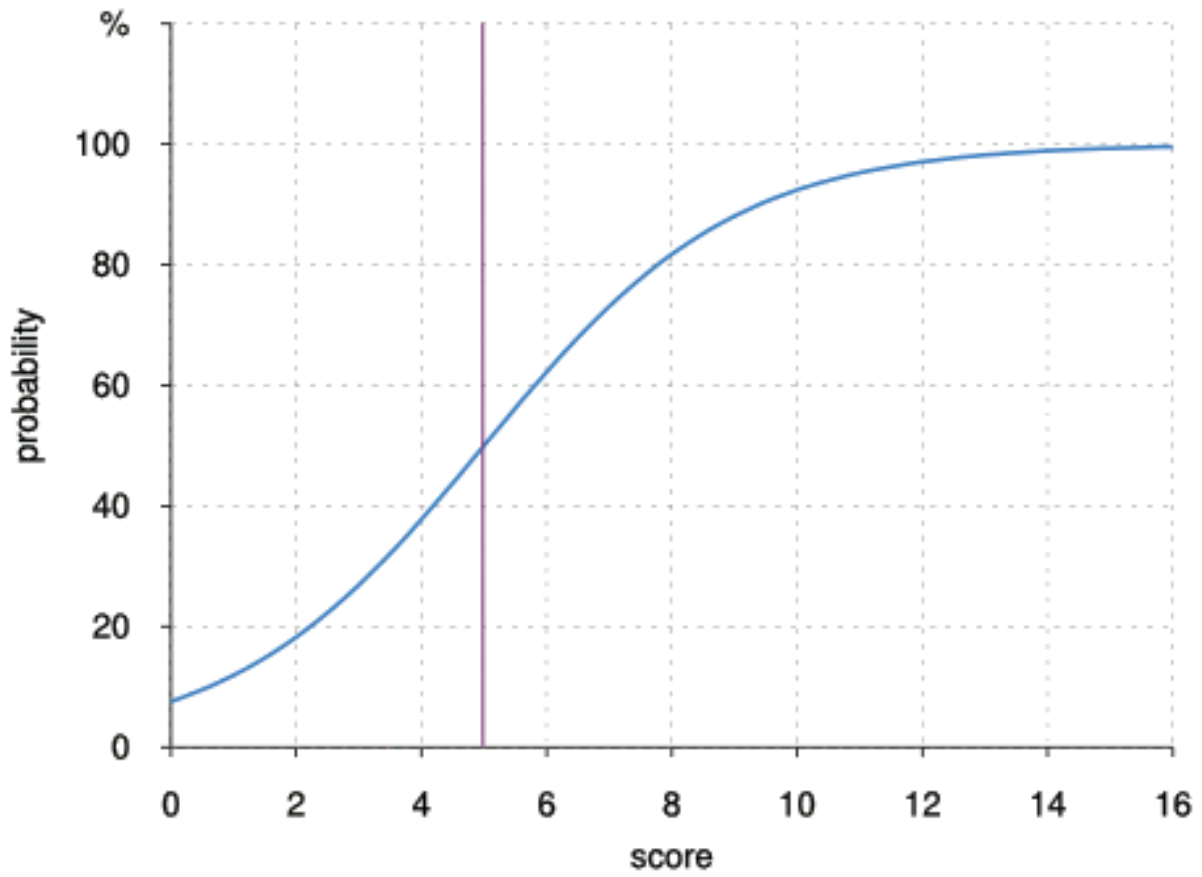
[dnsbl.conf](#)

Test Scores

Each test has a corresponding weight. All tests are run against each message; for each test that matches the message, the weight associated with the test is added to the total spam score for the message.

Test scores can be specified as a numerical value (`score`), as a percentage (`probability adjustment %`), or both. Scores and probability adjustments can be specified to multiple decimal places, although the PureMessage Manager Anti-Spam Rules page only displays three decimal places. Scores and probability percentages can be specified as either positive or negative values; negative values are prefixed with a minus symbol. Positive scores and probability percentages increase the likelihood that a message is classified as spam; negative values decrease the likelihood.

The chart below shows the relationship between scores and percentages. A score of "5" results in a 50% spam probability.



If an anti-spam rule has both a score and a probability percentage, the score is converted to a percentage, and the probability percentage is added to the result. For example, if a rule has a score of 5 and a probability percentage of 10%, the score is converted to a percentage (50%), and the probability percentage is added, resulting in a total score for the rule of 60%.

The PureMessage policy script is configured to perform actions on each message based on the message's total spam score. For example, the default policy adds an `X-PerlMx-Spam` header to messages with a spam probability. If the message's spam probability exceeds 50%, it adds the `X-PerlMx-Spam` header, and also alters the subject line and copies the message to the quarantine.

Related concepts

[Managing Anti-Spam Rules](#) (page 124)

[Policy Overview](#) (page 242)

Configuring Spam Detection

Anti-spam configuration options determine the general functioning of spam detection within PureMessage. In most cases, PureMessage will provide good catch rates without any customization. However, you can maximize the effectiveness of PureMessage by ensuring that DNS settings are configured correctly, that trusted IP relays are specified, and that some form of IP blocking is enabled. It is also recommended that you specify "safe" character sets.

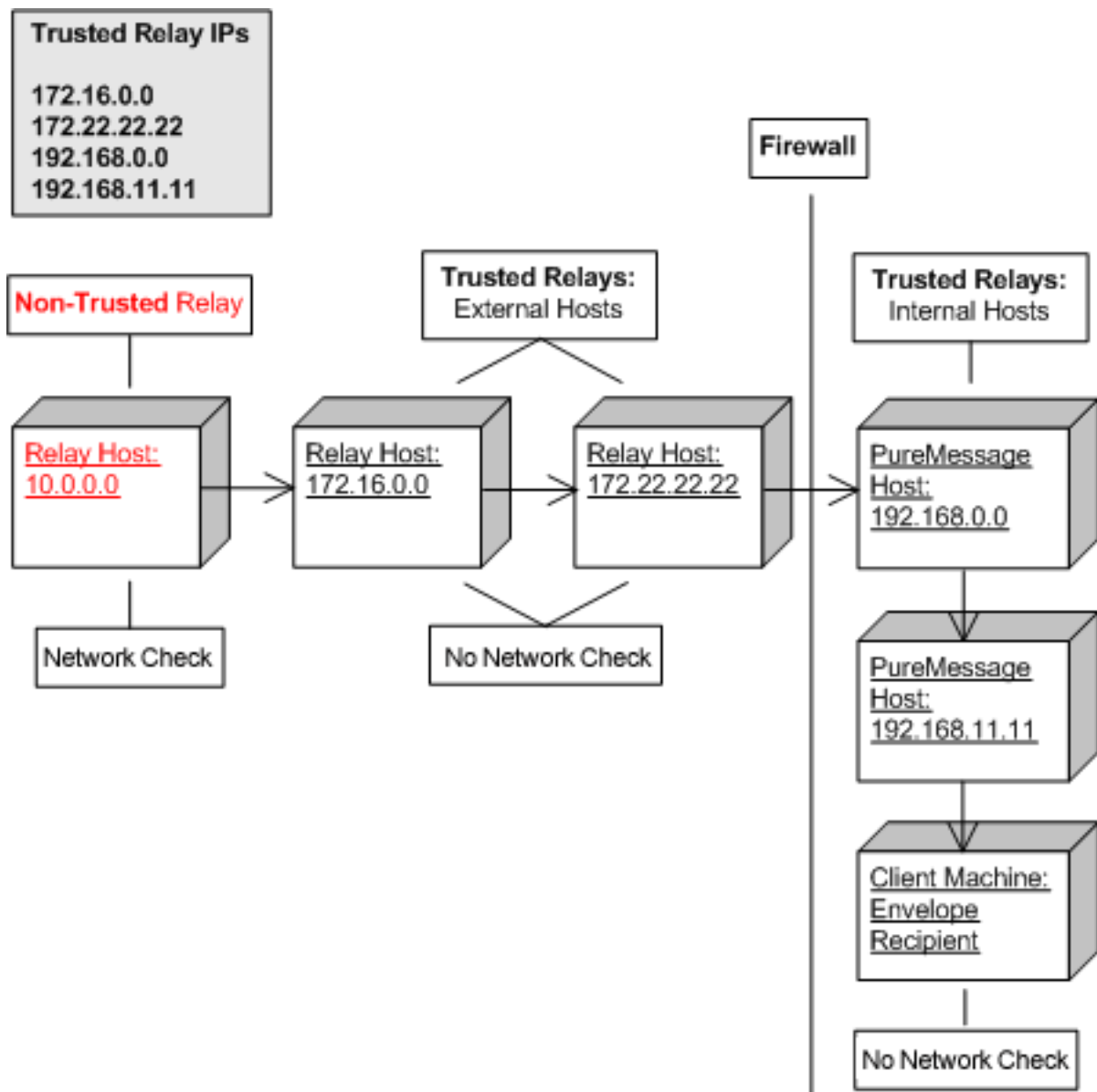
Optimizing DNS Checks

PureMessage performs a variety of DNS checks, including reverse DNS look-ups and queries that are handled via the Sophos SXL architecture. PureMessage performance is strongly affected by the connection speed between PureMessage and the DNS server. For optimal performance, install a local caching DNS server. Although it is possible to disable network checks completely by setting the `local_tests_only` option in `/opt/pmx6/etc/spam.conf` to "on", this is not recommended because it will negatively affect catch rates. You can specify the server(s) used for DNS based checks in `/opt/pmx6/etc/spam.d/net.conf`. The `net.conf` configuration file also allows you set values such as "retry" and "timeout".

Specifying Trusted Relays

Trusted relays are internal and external mail-filtering hosts that are known to be safe. Before an email message reaches its envelope recipient, it travels through a number of message-handling hosts that receive the message, and passes it to the next message-handling host on the internet. The message is relayed along this chain of hosts until it reaches its final destination, the envelope recipient.

DNS checks work in conjunction with the "trusted-relays" list, located in `opt/pmx/etc/`. This list should include all internal mail-filtering servers, and known, trusted external servers (for example, internet service provider (ISP) mail exchange servers). Configuring the "trusted-relays" list ensures that these IP addresses are exempt from the DNS checks. Only IP addresses (not domain names) can be entered in the "trusted-relays" list.



Relays with IP addresses within the 127.*.*, 192.168.*.* and 10.*.* blocks are always treated as internal relays. By default, the IP address of the first "external" relay is tested. All IP addresses of relays that are known to be safe, but are not included in the IP address blocks described above, should be added to the **Trusted Relay IPs** list. For example, if an ISP provides message-relay services for your company, the IP address of the ISP's mail server should be included in the **Trusted Relay IPs** list.

Populate the **Trusted Relay IPs** list via the Manager or at the command line. For more about configuring trusted relays, see "Configuring Anti-Spam Options" and "Editing Lists" in the *Manager Reference*. At the command line, edit the `trusted-relays` file, located by default in `opt/pmx/etc`.

Once the **Trusted Relay IPs** list is populated, configure the **Disable non-relay checks?** option on the **Policy > Anti-Spam Options** page in the PureMessage Manager. When the **Disable non-relay checks** is set to "Yes", only the first external relay is tested; checks of other relays in the receiving chain of relays are disabled, which can improve performance and reduce false positives.

Configuring IP Blocking

Policy-level IP blocking is configured by default in PureMessage. If you are able to position PureMessage at the outer edge of your network, it is recommended that you enable MTA-level IP blocking instead of policy-level blocking for improved performance. For instructions on enabling the IP Blocker Service, see "Enabling or Disabling MTA IP Blocking" in the Local Services Tab section of the *Manager Reference*.

Note

If your network has trusted local SMTP relays that pass inbound messages to the PureMessage, use policy-level blocking instead of MTA-level blocking, and add the local inbound SMTP relays to the Trusted Relays list. MTA-level blocking will only work correctly if PureMessage receives messages directly from the internet.

Configuring Safe Character Sets

It is recommended that you specify "safe" character sets. Several anti-spam rules analyze the character set of the message because foreign characters frequently indicate spam. Messages containing text in character sets that are identified as "safe" are exempted from these anti-spam rules. The default "safe" character set is read from the system's `LANG` environment variable.

Related concepts

[Managing Anti-Spam Rules](#) (page 124)

[PureMessage Default Lists \(see Trusted Relay IPs\)](#) (page 118)

Related tasks

[Configuring Anti-Spam Options](#) (page 127)

[Enabling or Disabling MTA IP Blocking](#) (page 178)

[Editing Lists](#) (page 120)

Related information

[net.conf](#)

[spam.conf](#)

3.5.3 Viruses

PureMessage uses the Sophos Anti-Virus engine. Scanning options are configured via the PureMessage Manager or the `sophos.conf` configuration file.

The PureMessage applications and configuration files used to configure anti-virus scanning from the command line are:

- `/opt/pmx6/etc/init.d/pmx-vscan`: A virus scanning service.

Anti-Virus Policy Related Configuration Files

- `/opt/pmx6/etc/virus.conf` : Specifies the action that should be performed if the virus engine fails.
- `/opt/pmx6/etc/virus.d/cantscan.conf` : Specifies what action `pmx_virus_clean` should take if `pmx_virus` fails to scan a message attachment.

- `/opt/pmx6/etc/virus.d/sophos.conf` : Contains the most commonly used Sophos Anti-Virus options.

Related information

[pmx-vscan](#)
[virus.conf](#)
[cantscan.conf](#)
[sophos.conf](#)

Unscannable Attachments

Some message attachments and message parts cannot be scanned for viruses because of corruption, encryption or missing parts.

Sophos Anti-Virus treats these messages as potentially dangerous and returns an internal failure code to the policy engine (`SOPHOS_SAVI_FILE_ENCRYPTED`, `SOPHOS_SAVI_FILE_CORRUPT` or `SOPHOS_SAVI_FILE_PART_VOL`). By default, it treats the message as if it were infected.

The method by which these scan failures are handled is determined by the settings in the `cantscan.conf` configuration file. Within this configuration file, a failure template is defined in the template setting (`cantscan.tmpl` by default). When message scanning fails, the template is used to construct a message to the original recipient advising that the original message was not delivered. The template file can be edited, or a custom template can be created. The following template variables can be used within the template:

- `%%DESC%%`: Expands to the description configured in `cantscan.conf`.
- `%%QID%%`: Expands to the Queue ID of the message.
- `%%ADMIN_ADDRESS%%`: Expands to the PureMessage administration address, as defined in the admin setting in the `pmx.conf` configuration file (or on the **Edit Global Options** page on the **Local Services** tab in the PureMessage Manager).

Related concepts

[PureMessage Logs](#) (page 300)

Related tasks

[Setting Global Options](#) (page 176)

Related information

[sophos.conf](#)
[cantscan.conf](#)
[pmx.conf](#)

Uncleanable Attachments

In a similar manner, the anti-virus engines provide support for "cleaning" (that is, removing a virus from a message before sending it to the intended recipients). If the engine is unable to clean the virus from a message, it uses the `cantclean.tmpl` template file to generate a message to the original message recipients. This template file can be altered. Use the following template variables within the template:

- `%%VIRUS_IDS%%`: Expands to the engine's identification code for the virus.
- `%%QID%%`: Expands to the Queue ID of the message.
- `%%ADMIN_ADDRESS%%`: Expands to the PureMessage administration address, as defined in the admin setting in the `pmx.conf` configuration file (or on the **Edit Global Options** page on the **Local Services** tab in the PureMessage Manager).

Related concepts

[PureMessage Logs](#) (page 300)

Related tasks

[Setting Global Options](#) (page 176)

Related information

[sophos.conf](#)

[pmx.conf](#)

[virus.conf](#)

Updating Virus Heuristics

A default scheduled job automatically updates virus definition files. Refer to the **Scheduled Services** section on the **Local Services** tab in the PureMessage Manager to view or change this configuration.

To manually check for virus definition file updates, use the **Available Updates** page on the **Support** tab in the PureMessage Manager.

Related concepts

[Local Services Tab](#) (page 152)

[Support Tab: Available Updates](#) (page 189)

Related tasks

[Managing Scheduled Jobs](#) (page 172)

3.6 Quarantine Administration

This section describes management of the PureMessage quarantine, a temporary holding place for messages that are deemed potentially problematic by the PureMessage policy. Quarantined messages can then be reviewed, and released or deleted.

Policy settings determine which messages are quarantined. For example, rules can be configured to quarantine messages if their spam probability exceeds a certain level.

Quarantined messages are managed using either the `pmx-qman` command-line program or the graphical Quarantine Manager. Depending on PureMessage installation options, end users can manage their own quarantined messages using the End User Web Interface.

Note

You can also view and manage the quarantine using the options available via the **Search** tab of the PureMessage Groups Web Interface. Although it is primarily used to delegate tasks under the group administration model, the Groups Web Interface can be configured as a quarantine manager. You might consider this alternative if you want to take advantage of some of the special search features available only in the Groups Web Interface, and you are comfortable managing PureMessage with multiple interfaces. See the “Administrative Groups” section of the *Administrator’s Reference* for more information.

Quarantine Digests alert recipients that messages have been quarantined. By replying to digests, users can release their messages from the quarantine.

This section includes descriptions of operations that can only be done from the command line. It also describes a number of scheduled quarantine-related tasks.

Related concepts

[Policy Configuration](#) (page 242)

This section describes the customization and fine tuning of message filtering for viruses, spam, and organizational policy compliance.

[Quarantine Tab](#) (page 128)

[End User Management](#) (page 294)

This section describes the options that can be made available to PureMessage end users, the people within your organization who are the senders and recipients of email that is processed by PureMessage.

[Administrative Groups](#) (page 201)

This section describes the setup and Management of the Groups Web Interface, which a system administrator can use to delegate selected tasks to other system administrators.

[Digests Management](#) (page 286)

Related information

[pmx-qman](#)

3.6.1 Quarantine Management

Typically, quarantined messages have an uncertain spam probability threshold (50% or greater), or they contain viruses that may be removable but should be processed manually, or they are encrypted, or they contain unscannable attachments that should be manually examined.

- `/opt/pmx6/bin/pmx-qman` : The PureMessage quarantine manager.
- `/opt/pmx6/bin/pmx-qindex` : Makes new messages in the PureMessage quarantine available.
- `/opt/pmx6/bin/pmx-qmeta-index` : Indexes the metadata in quarantined messages for faster searching.
- `/opt/pmx6/bin/pmx-qsearch` : Searches for messages in the quarantine.
- `/opt/pmx6/bin/pmx-queue` : Manages queued messages.
- `/opt/pmx6/bin/pmx-qexpire` : Expires messages in the PureMessage quarantine.
- `/opt/pmx6/bin/pmx-qrelease` : Iterates through the centralized storage of requests for approvals, forwards or deletes quarantined messages, and processes them. Usually run as a scheduled service.

Related Configuration Files

- `/opt/pmx6/etc/pmx.d/pmdb.conf` : Specifies the location of the quarantine store, “pmx_quarantine”, the type of the quarantine (directory or database) and the information needed to connect to the database if that type is selected. Also sets whether this store or some other is used for resources and reports data.

Related information

[pmx-qman](#)

[pmx-qindex](#)

[pmx-qmeta-index](#)

[pmx-qsearch](#)

[pmx-queue](#)

[pmx-qexpire](#)

[pmx-qrelease](#)

[pmdb.conf](#)

Quarantine Directories and Files

Directories

During installation, PureMessage creates a directory structure for storing quarantined messages. This structure is created, by default, in the `var/qdir` directory beneath the PureMessage installation directory. Subdirectories are created beneath the `qdir` directory to store messages as they pass through various stages of quarantine processing.

- `var/qdir/counters`: Uses a series of files to keep a count of the various actions performed by quarantine utilities.
- `var/qdir/tmp`: Stores messages while they are processed by the policy engine.
- `var/qdir/cur`: When the `pmx-qman` command-line Quarantine Manager, or the graphical Quarantine Manager is invoked, messages in the `var/qdir/new` directory are indexed and moved to the `var/qdir/cur` directory.
- `var/qdir/sent`: Stores messages that have been approved and delivered to the intended recipient.
- `var/qdir/test`: Stores messages generated by the `pmx-test` command-line program or the Policy Test function in the Manager.
- `var/qdir/trash`: Stores messages that have been deleted using the `pmx-qman` command-line Quarantine Manager, or the graphical quarantine manager. This directory is not created until messages are deleted.

Numbered Message Directories

Once quarantined messages are indexed, they are stored in the `cur` directory as individual files with numeric filenames equivalent to the message's Quarantine ID. Under the `cur` directory are a series of numbered directories. Each numbered directory stores the individual message files with corresponding numeric data. The first directory describes the number of digits in the filename. The next directory describes the first number (or series of numbers) in the filename. The following list illustrates where messages are stored:

```
4/1/1000 ... 1999
4/2/2000 ... 2999
...
4/9/9000 ... 9999
5/10/10000 ... 10999
5/11/11000 ... 11999
...
5/99/99000 ... 99999
...
9/999/999/999999000 ... 999999999
```

A message with filename 1234, for example, is stored in the `4/1/` subdirectory. A message with filename 57453 is stored in the `5/57/` subdirectory.

Related concepts

[Quarantine Tab](#) (page 128)

[Testing Policies](#) (page 270)

Related information

[pmx-qman](#)

pmx-test

Quarantine Indexing

Metadata from the quarantined messages (sender, recipient, date, subject, etc) is stored in a database to speed up queries of the quarantine. PureMessage supports two database formats for these indexes. PostgreSQL, an object-relational database system, is the default PureMessage database. CDB is a simple flat-file database, which is suitable for smaller installations running on a single PureMessage server.

- **PostgreSQL:** PostgreSQL is a full-fledged relational database. It is required for installations using Centralized Quarantine Management and Reporting and is recommended for installations using the End User Web Interface.
- **CDB:** CDB (a package for creating and reading constant databases) is a flat-file database back end for storing quarantine metadata. A CDB-indexed quarantine can scale to millions of messages, is usually less work to administer and is much faster to index than PostgreSQL. Note, however, that the reporting functionality in PureMessage relies on PostgreSQL. You can manually re-index the database back end by running the command:

```
pmx-quarantine reindex --index-type=cdb
```

When using CDB, the Quarantine ID, Queue ID (as designated by the MTA), recipient and sender fields are indexed. Searching for an exact value in any of these fields results in fast search results. If a wildcard is specified in an address search, no index is used, resulting in a slower search (commensurate with the number of messages in the quarantine).

When using PostgreSQL, information from the Subject, Body and other message parts are also indexed to provide fast queries and message previews in the End User Web Interface and the Manager interface to the quarantine. Advanced queries or queries involving wildcards are faster than with CDB.

Related concepts

[Centralized Quarantine Management](#) (page 285)

[Reports Tab](#) (page 148)

Related information

[CDB web site](#)

Changing the Quarantine indexing database

PureMessage uses a database to index fields from quarantined messages to enable faster searching. If you are changing or upgrading your PureMessage installation, you may wish to change the database used for this quarantine data for reasons outlined in [Quarantine Indexing](#) (page 283).

To change from PostgreSQL to CDB:

```
pmx-database stop
pmx-quarantine reindex --index-type=(bdb or cdb)
pmx-config quarantine_type dir
```

Remove the `pmx_db` line from `pmx.conf` (located by default in `/opt/pmx6/etc`), then run `pmx-qmeta-index` or allow it to run automatically from the Scheduler. For more information, see the Local Services Tab section of the *Manager Reference*.

To change from CDB to PostgreSQL:

Run the `pmx-postgres-enable` script as the “pmx6” user.

After PostgreSQL is configured, use the following procedure to transfer the current quarantine data to the PostgreSQL database. Note that it takes approximately one hour to index 125,000 messages.

1. Disable the `pmx-qmeta-index` scheduled service.
2. Kill any running `pmx-qmeta-index` services.
3. While logged in as the pmx user (by default, “pmx6”), enter `pmx-quarantine reindex`. When reindexing is complete, enter `pmx-qmeta-index`.
4. Re-enable the `pmx-qmeta-index` scheduled service.

Related concepts

[Quarantine Indexing](#) (page 283)

[Local Services Tab](#) (page 152)

PureMessage PostgreSQL Command-Line Programs

PureMessage has command-line programs used for specific PostgreSQL-related tasks. The following commands are available:

- `/opt/pmx6/etc/postgres/pmx-pg-tune` : Optimizes the shared memory usage for PureMessage PostgreSQL.
- `/opt/pmx6/etc/postgres/pmx-pg-migrate` : Migrates PostgreSQL data as part of the process of upgrading from PostgreSQL 7.4 to 8.3.
- `/opt/pmx6/etc/postgres/pmx-pg-switch` : Completes the process of migrating from PostgreSQL 7.4 to 8.3 by importing data into the new database and integrating PureMessage with the new version of PostgreSQL.

Related information

[pmx-pg-tune](#)

[pmx-pg-migrate](#)

[pmx-pg-switch](#)

Consolidated vs. Centralized Quarantines

Quarantine consolidation (where messages are moved from one or more source quarantines to a central quarantine) was once the preferred method of managing quarantined messages on multiple PureMessage servers. Although this functionality is still supported, Centralized Quarantine Management (where messages are kept in their original quarantines, but metadata is stored in a central database), is a more efficient alternative.

Consolidating Quarantined Messages from Multiple Servers

It is possible to consolidate the quarantined messages from multiple PureMessage servers on one server, allowing you to administer all quarantined messages from a single location.

Note

If you configure a consolidated quarantine in conjunction with quarantine digests, a single digest is generated for each end user, rather than separate digests from each of the quarantines.

To consolidate quarantined messages, you must be able to log on to the server where messages are to be stored from the machine(s) where the messages are originally quarantined via a secure shell, as the PureMessage user (by default "pmx6").

Run the following command to export quarantined messages from one of your PureMessage servers to the server where the messages will be consolidated:

```
pmx-qindex; PMX_MESSAGE_STORE=dir:/opt/pmx6/var/qdir pmx-store-export --
importer\
  'ssh pmx@ConsolidatedQuarantineServer PMX_MESSAGE_STORE=dir:/opt/pmx6/
var/qdir\
  /opt/pmx6/bin/pmx-store-import -';rm -f /opt/pmx6/var/qdir/cur/
index_log
```

The above command assumes that you accepted the default username (pmx6) and default installation directory (/opt/pmx6) when you installed PureMessage on the machine that will store the quarantined messages.

To automate the consolidation of quarantined messages, add the above command as a scheduled job. You must configure the "pmx6" user to log on to the consolidated quarantine server without manual intervention. Therefore, you will have to configure SSH keys on both the consolidation server and the servers where the messages are first quarantined. See the SSH documentation for detailed instructions.

Be sure to include the final portion of the command (rm -f /opt/pmx6/var/qdir/cur/index_log). This prevents the index_log file from growing and consuming excessive disk space.

The pmx-qmeta-index scheduled job should be disabled on all servers, except the server where messages are consolidated (see "Managing Scheduled Jobs" for more information).

Related concepts

[Digests Management](#) (page 286)

[Digest Configuration](#) (page 287)

Related tasks

[Managing Scheduled Jobs](#) (page 172)

Related information

[pmx-store-export](#)

[pmx-store-import](#)

Centralized Quarantine Management

Centralized Quarantine Management provides the ability to keep metadata about quarantined messages in a single database (PostgreSQL). It is more efficient than quarantine consolidation because it does not transfer messages in their entirety to a central server. Only metadata about messages is transferred (such as the message's "envelope to" and "envelope from" data). The entire message is only transferred when approved using the End User Web Interface, the Quarantine Manager, pmx-qman, or pmx-qdigest-approve.

Example implementations of centralized quarantines can be found in the Deployment Strategies.

Related concepts

[Deployment Strategies](#) (page 6)

[Managing the PostgreSQL Service](#) (page 166)

Related information

[pmx-postgres-enable](#)

Managing Quarantined Messages

Quarantined messages are managed using either the `pmx-qman` command-line program or the graphical Quarantine Manager. Either interface can be used to search for quarantined messages, and to perform various tasks with those messages.

- **Approving Quarantined Messages:** When a message is approved, it is forwarded to the intended recipient and moved to the `var/qdir/sent` directory. (Alternatively, users can approve their own messages through the use of Quarantine Digests or the End User Web Interface.)
- **Forwarding Quarantined Messages:** When a message is forwarded, it remains in the quarantine but is also sent to the specified recipient.
- **Deleting Quarantined Messages:** Messages in the quarantine can be “deleted” using the quarantine management interface. This removes them from the `var/qdir/cur` directory and moves them to the `var/qdir/trash` directory. These messages are still part of the quarantine, although they are not displayed during quarantine searches (or included in quarantine digests). To permanently delete or archive messages in the quarantine, see “Expiring Quarantined Messages”.
- **Saving Quarantined Messages:** When a message is saved, you are prompted to supply a file location. The message is exported as an “.mbox” file. The original message remains in the quarantine.

Related concepts

[Quarantine Tab](#) (page 128)

[Digests Management](#) (page 286)

Related information

[pmx-qman](#)

Expiring Quarantined Messages

The `pmx-qexpire` program deletes messages in the `var/qdir/Sent`, `var/qdir/Test`, `var/qdir/cur`, and `var/qdir/trash` directories. Configuration options are set in `pmx/etc/pmx.d/quarantine_expire.conf`.

Related concepts

[Scheduling Automatic Quarantine Tasks](#) (page 292)

[Quarantine Tab](#) (page 128)

Related information

[pmx-qman](#)

[pmx-qexpire](#)

3.6.2 Digests Management

Quarantine Digests are messages sent to end users by PureMessage that contain a list of users' quarantined messages. By replying to their digests, users can release (approve) messages held in the quarantine.

Note

Email users should be advised to respond to quarantine digests in a timely fashion. Since quarantine digests and the quarantined messages themselves are both eventually expired, users must release messages prior to expiration. The length of the availability period will vary, depending on how you have configured the `pmx-qdigest-expire` and `pmx-qexpire` scheduled jobs. You can add a reminder to digests by editing the digest templates in `/opt/pmx6/etc/templates`.

After configuring quarantine digests (specifying the digest recipients, digest template, etc.) three PureMessage programs are used to administer the digest process. These are described in the “Generating Quarantine Digests” section.

If the End User Web Interface (EUWI) is installed, end users can use quarantine digests to automatically log in to the EUWI.

See “Troubleshooting Quarantine Digests” in the PureMessage FAQ for help with resolving digest problems.

- `/opt/pmx6/bin/pmx-qdigest-init` : Initializes digest generation from centralized quarantine.
- `/opt/pmx6/bin/pmx-qdigest` : Generates digests of quarantined email.
- `/opt/pmx6/bin/pmx-qdigest-approve` : Queues and delivers quarantined messages.
- `/opt/pmx6/bin/pmx-qdigest-expire` : Expires pending quarantine digests.

Related Configuration Files

- `/opt/pmx6/etc/pmx-qdigest.conf` : Contains configuration options for the `pmx-qdigest`, `pmx-qdigest-approve` and `pmx-qdigest-expire` programs. See the man pages for these respective programs for details.

Related concepts

[Generating Quarantine Digests](#) (page 289)

Related tasks

[Troubleshooting Quarantine Digests](#) (page 290)

Related information

[pmx-qdigest-init](#)

[pmx-qdigest](#)

[pmx-qdigest-approve](#)

[pmx-qdigest-expire](#)

[pmx-qdigest.conf](#)

Digest Configuration

General digest configuration includes:

- **Approval Address:** Specifies the sendmail alias that is used for message approvals (by default, `pmx-auto-approve@yourdomain.com`). If you are using a version of sendmail other than the one distributed with PureMessage, you must manually add this alias. See “Configuring an Existing Sendmail Installation” in the *PureMessage Installation Guide* for instructions.
- **Digest Expiry:** When the `pmx-qdigest-expire` program is run, it checks the digest expiry setting and deletes digests older than the specified number of days.

A digest configuration includes the following components:

- **Digest Template:** Digests are generated according to the contents of a template file. Template files must be located in the `etc` directory, beneath the PureMessage installation directory. Templates can be modified as required. For guidelines regarding custom digest templates, including available template variables and digest fields, see the `pmx-qdigest` man page.
- **Reason:** When a message is quarantined, the quarantine record contains the reason the message was not delivered, such as "spam" or "virus". Digests are generated for messages that match the specified reason.
- **Address List:** Digests are associated with lists of users (by default, the Quarantine digest users list). Custom lists of users can be configured; the ID code for custom digest lists must begin with the word "digest".

Related concepts

[Configuring an External Sendmail Installation](#) (page 45)

[Configuring an External Postfix Installation](#) (page 49)

Related information

[pmx-qdigest-expire](#)

[pmx-qdigest](#)

Consolidating Digests Generated for a Single User

By default, quarantine digests are generated for each email account that has messages addressed to it in the quarantine. However, some users have multiple email accounts and prefer that all quarantined messages for all accounts be included in one quarantine digest. This is accomplished through address mapping.

To consolidate digests:

1. Enable consolidated digests by editing the `pmx-qdigest.conf` configuration file and changing the `consolidate` option to "merged".
2. Configure address mapping by editing the `notifications` configuration file, located by default in the `etc` directory beneath the PureMessage installation. Create an entry that maps all the addresses you want consolidated into a single digest on the left side, and the address of the digest recipient on the right. Use spaces to separate multiple addresses. Use a colon to separate the addresses that you wish to consolidate from the digest recipient address

Note

The digest recipient specified on the right side must also be included (either explicitly or as part of a wildcard match) in the Quarantine digest users list for the consolidated digest to be sent.

Related concepts

[PureMessage Default Lists \(see "Quarantine Digest Users"\)](#) (page 118)

Centralizing Quarantine Digests

In multi-server deployments with a central server and one or more edge servers, digests are, by default, sent from individual edge servers to end users. In this configuration, end users receive a digest from each edge server that has quarantined messages addressed to the end user.

Alternatively, if you have configured a Centralized Quarantine using PostgreSQL, digests can be generated from a single server. In this configuration, end users receive a single digest from a single server that includes their quarantined messages from all edge servers. Quarantine digests can be disabled (that is, removed from the Scheduler) on the remaining servers.

If the quarantine digests previously ran only on edge servers, the new centralized digest server must be configured. Configure the server as per the instructions in [Digest Configuration](#), or copy the digest configuration (`pmx-qdigest.conf`, the digest-users list, and any customized digest templates) from an edge server.

To enable centralized digests, edit the `pmx-qdigest.conf` configuration file and change the centralized option to “true” and run the command:

```
pmx-qdigest-init --central
```

The `pmx-qdigest-init` utility initializes centralized digest storage. When the `pmx-qdigest` scheduled job is next run, messages quarantined since the last local digest was sent will be included. This utility must be run on each edge server that previously generated local digests.

Related concepts

[Centralized Quarantine Management](#) (page 285)

[Digest Configuration](#) (page 287)

Related tasks

[Managing Scheduled Jobs](#) (page 172)

Related information

[pmx-qdigest.conf](#)

[pmx-qdigest-init](#)

[pmx-qdigest](#)

Generating Quarantine Digests

Three programs are used to generate and administer quarantine digests. They are: `pmx-qdigest`, `pmx-qdigest-approve` and `pmx-qdigest-expire`. These programs are run automatically as scheduled jobs. See “Scheduling Digest Tasks” for the recommended settings for the scheduled jobs.

These scheduled jobs are used to automatically perform the following tasks:

- **Generate Digests:** The `pmx-qdigest` program analyzes quarantined messages and generates quarantine digests. It functions according to the configuration in the `pmx-qdigest.conf` configuration file; `pmx-qdigest.conf` settings can be manually edited, or administered on the **Digest Rules** page on the **Quarantine** tab in the PureMessage Manager. `pmx-qdigest` generates an email for each configured user that lists all the messages in the quarantine that were intended for that user.
- **End User Release Requests:** When users receive digests, they can release messages from the quarantine by replying to the digest. To release a specific message from the quarantine, users must click on the message’s ID, which generates a reply email containing the message ID. To release all messages listed in the digest, users must reply to the digest message.
- **Process End User Release Requests:** The `pmx-qdigest-approve` program responds to release requests from end users. When an end user requests delivery of a message in the quarantine, `pmx-qdigest-approve` releases the message and queues it for delivery.
- **Expire Pending Digests:** PureMessage stores a record of each quarantine digest. When an end user requests the release of a message, `pmx-qdigest-approve` validates the release request against the archived digest records. Periodically, (by default, every 5 days) the digest archive should be cleared. The `pmx-qdigest-expire` program clears archived digest records.

Note

Some email clients (such as Microsoft Outlook) have built-in message filtering that can be optionally enabled by the end user. These filters can result in quarantine digests being filtered by the end user's email client. Users can either disable their local mail client filter, or can adjust the filter to exclude PureMessage quarantine digests. See "Exempting Digests from Local Filters" for instructions on the latter.

Related concepts

[Scheduling Automatic Quarantine Tasks](#) (page 292)

[Scheduling Digest Tasks](#) (page 293)

[Managing Quarantine Digest Rules](#) (page 142)

Related tasks

[Managing Scheduled Jobs](#) (page 172)

Related information

[End User Documentation: Exempting Digests from Local Filters](#)

[pmx-qdigest.conf](#)

[pmx-qdigest](#)

[pmx-qdigest-approve](#)

[pmx-qdigest-expire](#)

Troubleshooting Quarantine Digests

This section provides information to assist with troubleshooting common problems encountered when generating quarantine digests.

Note

The commands shown below may vary depending on your operating system and PureMessage installation parameters (such as the installation path or quarantine location). Alter as necessary. All commands should be run as the PureMessage user ("pmx" by default).

Digests are not being generated

1. Verify that the following commands are running as scheduled jobs in the PureMessage Manager. Run the commands at the shell prompt and watch for errors.

<code>pmx-qindex</code>	(adds new messages to the PureMessage quarantine)
<code>pmx-qdigest -verbose</code>	(generates digests; provides verbose feedback)
<code>pmx-queue run</code>	(delivers digests in the message queue)

2. Run the following commands to verify that sendmail and PureMessage are running:

```
pmx status                (will return the status of PureMessage)

ps -uwx|grep sendmail    (FreeBSD or Linux: should show sendmail as a
                           running process)

...or...

ps -ef|grep sendmail      (Solaris: should show sendmail as a running
                           process)
```

3. Verify that sendmail and PureMessage are interacting by running the following command and watching for message activity.

```
pmx-mlog -verbose
```

This command reads the PureMessage message log. If there is no message activity, verify that a line similar to the following is in the `sendmail.mc` configuration file:

```
INPUT_MAIL_FILTER('Policy','s=inet:3365@localhost,F=T,T=C:5m;E:8m;R:4m;S:2m') dnl
```

If this line is present, but there is still no activity, verify that the proper sendmail binary is being used (for example, the distribution of sendmail included with PureMessage and not the system's native version of sendmail).

4. If the system has recently run out of disk space, compare the values from the following commands:

```
$ pmx-qdigest --dump|grep ^@

$ cat /opt/pmx6/var/qdir/counters/pmx-queue.cnt

$ cat /opt/pmx6/var/qdir/db.conf
```

The numbers returned by the first two commands and the value of the `last_id` setting in the `db.conf` should not vary more than the value in the `num_msg` setting in the `db.conf`. If they do, contact PureMessage Support and include the output from these three commands.

Some users are not receiving digests

1. Ensure that the user in question is a member of the **Quarantine digest users** list. View the list members on the **Policy** tab of the PureMessage Manager.
2. Verify that there are messages in the quarantine for the user in question that have not previously been included on a digest.

To check which messages have been included in digests, enter the following command on the command line:

```
pmx-qdigest --dump
```

Enter `man pmx-qdigest` for information about sorting and filtering output from this command.

Users cannot release quarantined messages

- When the user requests the release of a message from the quarantine by replying to a quarantine digest, verify that the message reaches the PureMessage server by checking the sendmail maillog. If digest replies from the end user are not reaching the PureMessage server, verify that the mail routing is correct.

- Check that the message being requested has not been deleted from the quarantine by the `pmx-qexpire` scheduled job. Compare the date of the message with the value of `expire_time` in `etc/pmx.d/quarantine_expire.conf`.
- Check that the quarantine digest itself has not expired by comparing the date of the digest with the value of `expire` in `/opt/pmx6/etc/pmx-qdigest.conf` (5 days by default).

Error Message: Can't call method `min_digest_id` on an undefined value

The following `pmx-qdigest` error may be caused by a corrupted quarantine scan database (`/opt/pmx6/var/qdigest/scan.db`).

```
Can't call method "min_digest_id" on an undefined value /opt/pmx6/bin/
pmx-qdigest line n
```

- Verify this with the following command:

```
pmx-qdigest --dump
```

If it does not return a list of quarantine IDs for each digest type and user, and instead returns an error like the one above, the `scan.db` file is almost certainly the cause.

- Check if the permissions and ownership of the file are correct:

```
permissions: -rw-r--r--
owner:      pmx
group:      pmx
```

- If these are incorrect, changing them with `chmod` or `chown` may solve the problem. If the permissions and ownership are correct, or if changing them back to the defaults does not solve the problem, generate a new `scan.db` file:

```
cd 'pmx prefix'/var/counters
mv scan.db scan.db.broken
pmx-qdigest --earliest <YYYY-MM-DD hh:mm:ss>
```

Set `<YYYY-MM-DD hh:mm:ss>` to the time of the earliest message to include in the digest. The quarantine scan will start from this point.

Once the `pmx-qdigest` command has completed, the `scan.db` database should be regenerated. The next quarantine digest will start with the last quarantine ID scanned during the run (in this case, the most recent message in the quarantine).

The next time `pmx-qdigest` is run by the Scheduler, it should complete normally.

3.6.3 Scheduling Automatic Quarantine Tasks

A number of automatic quarantine management tasks can be run as scheduled jobs. Some of these can be installed and configured during the PureMessage installation. Others, including the Quarantine Digest tasks, can be manually configured.

- **Index Quarantined Messages:** (`pmx-qindex`) To speed up message sorting and retrieval, this task indexes new messages in the quarantine.
- **Expire Quarantined Messages:** (`pmx-qexpire`) Delete or archive messages in the quarantine that are older than the specified number of days and hours.

Default quarantine digest Scheduler jobs are created (in a disabled state) during the PureMessage installation. These tasks can be configured and enabled on the command line by using `pmx-scheduler`, or the scheduled jobs section on the **Local Services** tab in the PureMessage Manager.

The programs scheduled to run automatically are described individually in “Generating Quarantine Digests”.

Related concepts

[Expiring Quarantined Messages](#) (page 286)

[Generating Quarantine Digests](#) (page 289)

Related tasks

[Managing Scheduled Jobs](#) (page 172)

Related information

[pmx-scheduler](#)

Scheduling Digest Tasks

Scheduling tasks or jobs can be done by using the PureMessage Manager (see “Scheduled Jobs”), by using the PureMessage command-line programs (see `pmx-scheduler`), or by directly editing the configuration files `etc/scheduler.conf` and `etc/scheduler.d/*.conf`.

The examples below show the default Scheduler settings for the programs used to generate and administer quarantine digests. It is assumed that the `pmx-queue` program (configured and enabled by default during installation) is running as a Scheduler job in order to handle the delivery of quarantine digests.

pmx-qdigest: The `pmx-qdigest` program generates quarantine digests. The default Scheduler configuration file for this program (`/opt/pmx6/etc/scheduler.d/pmx-qdigest.conf`) is as follows:

```
<event pmx-qdigest>
  desc = "Send out quarantine digests"
  type = exec
  action = "pmx-qdigest --quiet"
  enabled = 1
  <when>
    s = 0
    m = 0
    h = 15
  </when>
</event>
```

See the `pmx-qdigest` man page for a list of command-line switches.

pmx-qdigest-expire: The `pmx-qdigest-expire` program deletes digests older than the specified age. The default Scheduler configuration file for this program (`/opt/pmx6/etc/scheduler.d/pmx-qdigest-expire.conf`) is as follows:

```
<event pmx-qdigest-expire>
  desc = "Expire pending digests"
  type = exec
  action = "pmx-qdigest-expire --quiet"
  enabled = 1
  <when>
    s = 0
    m = 13
    h = 4
  </when>
</event>
```

See the `pmx-qdigest-expire` man page for a list of command-line switches.

Related tasks

[Managing Scheduled Jobs](#) (page 172)

Related information

[pmx-scheduler](#)

[pmx-qdigest](#)

[pmx-qdigest-expire](#)

3.6.4 Message Store Management

- `/opt/pmx6/bin/pmx-quarantine` : Invokes the `pmx-store` command for the quarantine message store.
- `/opt/pmx6/bin/pmx-store` : The message store manager.
- `/opt/pmx6/bin/pmx-store-import` : Imports messages from a PureMessage message store.
- `/opt/pmx6/bin/pmx-store-export` : Exports messages from the PureMessage message store.
- `/opt/pmx6/bin/pmx-store-expire` : Expires messages in PureMessage message store.

Related Configuration Files

- `/opt/pmx6/etc/data/resources/available.d/quarantine_expire.conf` : Contains the configuration settings for the `pmx-store-expire` program. Be sure to read the warning that appears on the `quarantine_expire.conf` man page that concerns manual changes to this file.

Related information

[pmx-quarantine](#)

[pmx-store](#)

[pmx-store-import](#)

[pmx-store-export](#)

[pmx-store-expire](#)

[quarantine_expire.conf](#)

3.7 End User Management

This section describes the options that can be made available to PureMessage end users, the people within your organization who are the senders and recipients of email that is processed by PureMessage.

PureMessage allows end users to manage messages via a web page. From the End User Web Interface (EUWI), users can create their own lists of Approved (whitelisted) Senders and Blocked (blacklisted) Senders, and manage their own quarantined messages.

The EUWI is administered using a variety of PureMessage Manager features. The HTTPD (RPC/UI) service runs the PureMessage End User Web Interface (EUWI). The status of this service is viewed on the **Local Services** tab, which also provides access to EUWI-related configuration options. Alternatively, the HTTPD (RPC/UI) service can be controlled and tested using the `pmx-httpd` and `pmx-rpc-enduser` command-line programs.

A list configured via the **Policy** tab determines which end users have access to the EUWI. See “Editing Lists” in the Policy Tab section of the *PureMessage Manager Reference* to change the pre-configured

list of approved end users. By default, on installation, all users can access the EUWI due to the “*” wildcard setting in the enduser-users list located under the `opt/pmx/etc` directory. To restrict user access, use email glob syntax matching in the enduser-users list. See “Email Globs” in the “Match Types” section of the *Manager Reference* for more information.

Many EUWI options are configurable via the **Quarantine** tab. See “Setting End User Options” in the *Manager Reference* to set the location and session options. See “Configuring End User Features” in the *Manager Reference* to configure end user access to specific components (for example, per-user whitelists). See “Managing End User Whitelists” and “Managing End User Blacklists” in the Quarantine Tab section of the *Manager Reference* to manage whitelists and blacklists for individual end users.

EUWI per-user list changes are synchronized to all PureMessage hosts. Add PureMessage hosts to the RPC list via the **Policy** tab. See the “RPC Hosts” entry in the “About PureMessage Default Lists” section of the Policy Tab documentation in the *Manager Reference* to configure the IP addresses of all PureMessage servers.

Related concepts

[Match Types](#) (page 115)

[Managing End User Lists](#) (page 139)

[PureMessage Default Lists \(see RPC Hosts\)](#) (page 118)

Related tasks

[Editing Lists](#) (page 120)

[Creating Lists or Maps](#) (page 114)

[Setting End User Options](#) (page 133)

[Configuring End User Features](#) (page 134)

Related information

[pmx-httpd](#)

[pmx-rpc-enduser](#)

3.7.1 Configuring Http/Https Access

By default, end users connect to the End User Web Interface (EUWI) over an https connection on port 28443. Optionally, you can configure PureMessage to permit unsecured EUWI access over http on port 28080 as well.

To allow EUWI access via both http and https:

At the command line, as the “pmx6” user, run the following command:

```
ln -sf /opt/pmx6/etc/manager/httpd2/ssl/default.conf /opt/pmx6/etc/manager/httpd2/ssl.conf
```

To reset to the default, https only, run the following command:

```
ln -sf /opt/pmx6/etc/manager/httpd2/ssl/http.conf /opt/pmx6/etc/manager/httpd2/ssl.conf
```

Note

The setting that you specify will also be used to access the Groups Web Interface on the same server.

3.7.2 Adjusting the GMT Offset for the End User Web Interface

In the quarantine, message dates and times are stored in GMT (Greenwich Mean Time). The value entered for the `gmt_offset` option in the `enduser.conf` file will be added to (or subtracted from) message's date and time so that the message is displayed in "local" time. For example, if the timestamp on a message is 9 AM GMT, and the value in this text box is set to "2h", the message time displayed in the End User Web Interface is 11 AM. Precede the entry with a minus symbol to subtract from GMT.

To change the GMT offset:

1. In the `/opt/pmx6/etc/enduser/enduser.conf` file, change the `gmt_offset` option to the desired value.
2. At the command line, as the "pmx6" user, run:

```
pmx-profile sync-to-db --resource=enduser_user_store --force
```

```
pmx-profile sync-to-db --resource=enduser_config --force
```

Related tasks

[Setting End User Options](#) (page 133)

3.7.3 Statically Linking to the End User Web Interface in a Digest

Email users served by PureMessage can access the End User Web Interface (EUWI) via quarantine digests when a static link to the EUWI is added in the `digest-spam.tpl` template.

Note

End users who have not previously authenticated with the EUWI must first request a password via the EUWI login page. See "Accessing the End User Web Interface" in the User Documentation for more information on end user authentication.

A session cookie is created once a user has authenticated with the EUWI. This cookie automatically authenticates users when the static EUWI link is selected from within the digest message.

To add a static link:

1. On the command line, navigate to the English templates directory (located, by default, in `/opt/pmx6/etc/templates/en/`).
2. Open the `digest-spam.tpl` template in your desired editor.
3. Add the enduser URL link to the template. The following example URL can be added to the `digest-spam.tpl` template: "Click this ``End User Web Interface`` link to log in to your account." To verify the EUWI link, go to the **Quarantine** tab in the Manager, and then click **End User Options**. The **End User URL** text box contains the link to your EUWI.
4. Save the `digest-spam.tpl` template.

Related information

[Accessing the End User Web Interface](#)

3.7.4 Disabling the End User Web Interface

If, for some reason, you decide to disable the End User Web Interface, stop the service on the **Local Services** tab of the PureMessage Manager, and then delete the `pmx-httpd` symbolic link from the `init.d` directory (located by default in `/opt/pmx6/etc`).

If you have disabled Quarantine Digests, follow the instructions in “Digests Management” in the *Administrator's Reference* to re-enable them.

Related concepts

[Digests Management](#) (page 286)

3.7.5 Adding Custom Graphics to the End User Web Interface

PureMessage allows you to replace the Sophos graphics in the End User Web Interface (EUWI) with custom graphics (for example, logos associated with an organization or company). These graphics will be displayed to email recipients who manage their quarantined messages through the EUWI.

To add a custom logo:

1. Log on to the EUWI server as the PureMessage user, “pmx6” by default.
2. Create a custom graphics directory within the `skins` directory that will be used to store custom logos. Run the following commands:

```
cd /opt/pmx6/lib/manager/HTTPD/www/skins/
cp -r default <CustomGraphicsDirName>
```

The `cp` command creates a new directory based on the `default` directory.

3. Go to the `/opt/pmx6/etc/enduser/enduser_ui.conf` file, and add the line:

```
skin = <CustomSkinName>
```

This configures the EUWI to use the new custom graphic when the HTTPD (RPC/UI) service restarts.

4. If an `enduser_ui.conf.ppmdist` file is found in the `/opt/pmx6/etc/enduser/` directory, move it to the home directory of the PureMessage user by running the following commands:

```
cd /opt/pmx6/etc/enduser/
mv enduser_ui.conf.ppmdist ~/enduser_ui.conf.ppmdist.backup
```

Alternatively, delete the file by replacing the `mv` command with:

```
rm enduser_ui.conf.ppmdist
```

5. [Optional] Customize logos as required, according to the following guidelines:
 - Use `.gif` files only.

- *Do not* modify any files or permissions of files in the `/opt/pmx6/lib/manager/HTTPD/www/skins/default/` directory.
- *Do not* modify the filenames in the `/opt/pmx6/lib/manager/HTTPD/www/skins/<CustomSkinName>/` directory; in other words, change the content but use the same filename.
- Keep the permission; compare against files under the `/opt/pmx6/lib/manager/HTTPD/www/skins/default/` directory.
- Someone with web design expertise can further customize the EUWI by modifying the `/background_image.css` file and `/main.css` file.

6. Finalize the customization by running these commands:

```
pmx-profile sync-to-db --resource=enduser_ui_config --force
pmx-httpd restart
```

7. Launch the EUWI to view the customization.

To revert to the default graphics for the EUWI, in the `/opt/pmx6/etc/enduser/enduser_ui.conf` file, set `skin = default`, and then perform steps 6 and 7 above.

3.7.6 User Documentation

PureMessage clients using the End User Web Interface (EUWI) have access to documentation that introduces basic PureMessage concepts and provides instructions for all options that are configurable by end users. While logged into the EUWI, users can click Help on the sidebar for assistance.

Related information

[How Does PureMessage Identify Spam?](#)

[PureMessage End User Web Interface](#)

[Accessing the End User Web Interface](#)

[Blocked Messages](#)

[Deleted Messages](#)

[Approved Senders](#)

[Blocked Senders](#)

[Options](#)

[Setting Rules in Your Email Client Software](#)

[Quarantine Digests](#)

[Exempting Digests from Client Software Rules](#)

3.8 Logs and Reports

This section describes the management of the log files that register activities and the activities that are logged. It also covers generation of reports that are drawn, in part, from this data.

PureMessage provides a variety of reports on performance, operations, and message-processing statistics as graphs and tables that can be scheduled for automatic generation and emailed to one or more specified recipients, as well as exported in CSV format for use in other applications. All of this can be done from the PureMessage Manager, although it can also be run from the command line.

Reports data is gathered from a variety of system and PureMessage log files and stored in the PostgreSQL database. The tasks that draw reports data from the logs and enter it into the database can be run as scheduled jobs that are enabled by default.

Log files can also be analyzed for security purposes, and PureMessage provides two features for this purpose. The Log Watch feature scans the PureMessage message log and reports on anomalies. The Log Monitor feature scans the `message_log` for specified entries and generates log entries of its own that can subsequently be analyzed and to which PureMessage can automatically react.

3.8.1 PureMessage Reports

Reports are summaries of various aspects of system operations, often drawn from system or PureMessage logs. Report data is stored in the PostgreSQL database, and reports can be viewed on demand or scheduled to be emailed.

System log reports are generated from the file specified in the `log_to` setting of the `pmx.conf` configuration file (by default `pmx_log`). These reports can be run as scheduled jobs or run manually from the command line.

- `/opt/pmx6/bin/pmx-reports-set-message-size-ranges` : If you need to recreate the reports database, you should run this once only to populate the size ranges of the “`prd_msg_size_range`” table.
- `/opt/pmx6/bin/pmx-reports-set-probability-ranges` : If you need to recreate the reports database, you should run this once only to populate the spam probability ranges of the “`prd_probability_range`” table.
- `/opt/pmx6/bin/pmx-reports-set-time-ranges` : Schedule this command to run frequently to populate the “`prd_period`” table with time ranges prior to running the `pmx-reports-consume-message-log`, `pmx-reports-consume-pmx-log` and `pmx-reports-consume-quarantine` commands.
- `/opt/pmx6/bin/pmx-reports-consume-message-log` : Schedule this command, or optionally run it manually, to ensure that the reports database is updated with the `message_log` data.
- `/opt/pmx6/bin/pmx-reports-consume-pmx-log` : Schedule this command, or optionally run it manually, to ensure that the reports database is updated with the `pmx_log` data.
- `/opt/pmx6/bin/pmx-reports-consume-blocklist-log` : Schedule this command, or optionally run it manually, to collect report data from the `blocklist_log`.
- `/opt/pmx6/bin/pmx-reports-consume-quarantine` : Schedule this command, or optionally run it manually, to ensure that the reports database is updated with the quarantine data.
- `/opt/pmx6/bin/pmx-reports-mailer` : Schedule this command, or optionally run it manually, to generate any of six possible reports, and send them to the specified address.
- `/opt/pmx6/bin/pmx-reports-mailer-v2` : If you are using the Groups Web Interface for reporting, schedule this command, or optionally run it manually to generate reports, and send them to the specified email address.

Related information

[pmx.conf](#)

[pmx-reports-set-message-size-ranges](#)

[pmx-reports-set-probability-ranges](#)

[pmx-reports-set-time-ranges](#)

[pmx-reports-consume-message-log](#)

[pmx-reports-consume-pmx-log](#)

[pmx-reports-consume-blocklist-log](#)

[pmx-reports-consume-quarantine](#)

[pmx-reports-mailer](#)
[pmx-reports-mailer-v2](#)

3.8.2 PureMessage Logs

Logs automatically record aspects of system activity in the form of text files, thus providing useful information on PureMessage system operations.

- `/opt/pmx6/bin/pmx-log` : Runs `tail` or `grep` on the PureMessage control program log.
- `/opt/pmx6/bin/pmx-log-summary` : Summarizes the PureMessage control program log.
- `/opt/pmx6/bin/pmx-logsearch` : Provides options for searching the PureMessage logs.
- `/opt/pmx6/bin/pmx-mlog` : Runs `tail` or `grep` on the PureMessage message log.
- `/opt/pmx6/bin/pmx-csl` : Manages PureMessage Central Server logs.
- `/opt/pmx6/bin/pmx-mark` : Writes timestamp marks in log files.

Related Configuration Files

- `/opt/pmx6/etc/logrotate.conf` : A configuration file for the `logrotate` utility (which is standard on some UNIX systems) that determines its handling of PureMessage log files.

Related information

[pmx-log](#)
[pmx-log-summary](#)
[pmx-logsearch](#)
[pmx-mlog](#)
[pmx-csl](#)
[pmx-mark](#)
[logrotate.conf](#)

Message Log Syntax

The message log stores information about each message processed by the PureMessage milter. The log file is specified in the `message_log` setting in the `pmx.conf` configuration file (by default `/opt/pmx6/var/log/message_log`).

The message log file contains one line of data for each message. The line begins with the date and time that the message was processed. The rest of the fields are key/value pairs separated with an “=”. Some fields may consist only of keys, rather than key/value pairs, in which case the “=” symbol is not present.

Keys:

- `a`: The milter status action for this message, and the callback that provided the final action.
- `action`: The PureMessage processing action performed on the message.
- `at`: Attachments (see “Adding Attachment Information”).
- `b`: blocklist reason (“reject” or “ok”); appears when the `pmx_blocklist` policy rule is used.
- `f`: The envelope-from address.
- `fur`: The IP address of the first untrusted relay.
- `h`: The spam rule name. Repeats if multiple spam rules are hit.
- `p`: The spam probability value for the message; a number between 0 and 1.
- `pmx_action`: Can contain as many as five parts, including the final action taken on the message, and the reason the action was taken. If the addressee belongs to a PureMessage group, the group

is listed next, followed by the individual recipient. If the recipient has an alias, this appears as the final part of the entry. If not, the recipient address is repeated. If any one of the five parts is not available or does not apply, a dash ("-") appears in its place.

- q: The sendmail queue id of the message.
- r: The hostname of the relay SMTP server.
- Size: The message size in bytes
- t: The envelope-to address. Repeats when there are multiple recipients of the message.
- tm: The time (in seconds) that it took to process the message.
- v: The virus ID. Repeats if multiple viruses are found.
- vs: Virus-scanned: records when a message has been scanned for viruses.

Return codes:

- a: accept
- c: continue
- d: discard
- r: reject
- t: tempfail

Event codes:

- eom: end of message
- eoh: end of headers
- connect: MTA connect
- abort: MTA abort

Example:

For example, a typical line from message_log which logs a message might look like this:

```
2007-01-27T16:48:58 q=i0S0miXk018339 f=<sender@domain.com>
t=<recipient@example.org> p=0.351 h=RCVD_IN_SBL h=EXCUSE_19
h=LINES_OF_YELLING h=__EVITE_CTYPE h=__CTYPE_CHARSET_QUOTED
h=__CT_TEXT_PLAIN h=__CT h=__HAS_MSGID h=FREE_MONEY h=__SANE_MSGID
h=NO_REAL_NAME h=__TO_MALFORMED_2 h=__MIME_TEXT_ONLY h=__MIME_VERSION
Size=2274 r=relay.someplace.net tm=1.80 a=a/eom
```

This message was sent from sender@domain.com to recipient@example.org at 16:48.58 on Jan. 27 2007 via relay.someplace.net. It was 2274 bytes in size, took 1.0 seconds to process, triggered a number of anti-spam rules, received a total Spam score of %35.1 and was accepted by the mailer at the end of message event.

Customizing Message Log Information:

Additional custom keys and key/value pairs can be written to the message log using the `pmx_mark` and `pmx_mark1` policy actions. (In the PureMessage Manager, these actions are called "Log the message with key/value pair" and "Log the message with keyword". See "Actions Defined" in the Policy Tab section of the *Manager Reference* for more information.) For example, the default version of the policy filter adds an "i" key to indicate messages that originated from local hosts, and a "Size" key with the value of the message size, in bytes. The `pmx-test` program adds a `Test-ID` key and value for the purpose of tracking the message.

Customizing Message Log Reporting:

A variety of third-party tools can be used to generate custom reports from the message log. In addition, the `pmx-mlog` program can be used in conjunction with `grep` options to extract data from

the message log. Use the `--verbose` option with `pmx-mlog` to display the log contents in multi-line format.

Related concepts

[Adding Attachment Information](#) (page 303)

[Actions Defined](#) (page 93)

Related information

[pmx.conf](#)

[Policy Actions](#)

[pmx-test](#)

[pmx-mlog](#)

Other Log File Syntax

Unlike `message_log`, `pmx_log` does not present each message as a one line summary but rather records debug, info, notice, warning and error messages from the `pmx-milter` process. The verbosity of this log is controlled by the `debug_level` setting in `pmx.conf`. The default `debug_level` is 0, which causes all log messages with “DEBUG” priority to be suppressed. Greater values (to a maximum of 9) increase the amount of information sent to `pmx_log`.

When debugging problems with PureMessage or looking for specific policy tests and actions, it is useful to have the `debug_level` set to at least 5. As this creates a very verbose log, change the `debug_level` back to 0 when finished troubleshooting or configure `logrotate` to rotate `pmx_log` frequently.

More information on using `logrotate` to manage PureMessage logs can be found in the “Rotating PureMessage Log Files” section of Installing PureMessage for UNIX.

Related concepts

[Rotating PureMessage Log Files](#) (page 38)

Related information

[pmx.conf](#)

[pmx-milter](#)

Preventing Reporting on Unscannable Data

To prevent the reporting of unscannable data, modify the default policy (`policy.siv`). This can be done on the **Policy** tab, or from command line. Modifications can be made to both the internal and external host check sections as follows:

Internal Host Check

```
# attr NAME=Reject mail containing viruses
if pmx_virus {
    # attr NAME=Check for certain error codes
    if not pmx_virus_id :comparator "i;ascii-casemap" \
        :matches
        ["SOPHOS_SAVI_FILE_ENCRYPTED","SOPHOS_SAVI_FILE_CORRUPT",\
         "SOPHOS_SAVI_FILE_PART_VOL"] {
        reject :rcode 550 "One or more viruses were detected in the
message.";
        pmx_mark "I-Virus" "%%VIRUS_IDS%%";
        stop;
    }
}
```

External Host Check

```
if pmx_virus {
    # attr NAME=Check for certain error codes
    if not pmx_virus_id :comparator "i;ascii-casemap" \
        :matches
        ["SOPHOS_SAVI_FILE_ENCRYPTED","SOPHOS_SAVI_FILE_CORRUPT",\
         "SOPHOS_SAVI_FILE_PART_VOL"] {
        pmx_quarantine "Virus";
        pmx_mark "E-Virus" "%%VIRUS_IDS%%";
    }
}
```

Note

Unscannable files (either encrypted or corrupt) may contain viruses.

Related concepts

[The Default PureMessage Policy \(page 243\)](#)

Adding Attachment Information

To add attachment information to the `var/log/message_log`, add the following line to the `etc/policy.siv` file:

```
pmx_attachment_log;
```

This change creates entries in the `var/log/message_log` similar to the following:

```
2006-02-09T14:12:51 q=gDBAJDaci7 f=
t=<somebody@YourCompany.com>
at=1,3483,multipart/alternative at=1,3056,text/html at=1,0,text/plain
p=0.436 h=FROM_NUM_AT_WEBMAIL h=RCVD_FAKE_HELO_DOTCOM
h=X_MSMAIL_PRIORITY_HIGH h=NO_REAL_NAME h=__AOL_FROM h=__CT
h=__CTYPE_HAS_BOUNDARY h=__CTYPE_MULTIPART h=__CTYPE_MULTIPART_ALT
h=__HAS_MSGID h=__HAS_MSMAIL_PRI h=__HAS_X_MAILER h=__HAS_X_PRIORITY
h=__LINES_OF_YELLING h=__MIME_HTML h=__MIME_VERSION h=__SANE_MSGID
h=__TAG_EXISTS_HTML h=__X_MSPRI_HI Size=5880 r=unknownhost tm=0.26
a=a/eom
```

There are four fields that make up each at key. For example, in the following log entry:

```
at=1,3056,text/html
```

- `at`: indicates an attachment
- `1`: indicates the number of attachments of this type
- `3056`: indicates the total size of all attachments of this type
- `text/html`: indicates the attachment type.

3.8.3 Log Watch Options

- `/opt/pmx6/bin/pmx-mlog-watch`: Scans the PureMessage message log and reports on anomalies.
- `/opt/pmx6/bin/pmx-mlog-react`: Reads `pmx-mlog-watch` reports and adds offenders to policy lists.

Related Configuration Files

- `/opt/pmx6/etc/logwatch.conf`: The configuration file for the `pmx-mlog-watch` command, for which the options can be specified either on the command line or in this file. See the documentation for the `pmx-mlog-watch` command for details.

Related information

[pmx-mlog-watch](#)

[pmx-mlog-react](#)

[logwatch.conf](#)

3.9 Troubleshooting

3.9.1 Why aren't messages being scanned for spam?

There are several things to check if it appears that spam is not being caught by PureMessage:

Was the message routed through the PureMessage server?

The default PureMessage policy adds an “X-PMX-Version” header to mail from external hosts. If the message does not have this header, it was not processed by PureMessage.

Is `F=T` set in sendmail's `INPUT_MAIL_FILTER` line?

The “F=” parameter in the `sendmail.mc` configuration file determines the action that sendmail takes if the `concurrency_limit` specified in the `pmx.conf` configuration file is exceeded or if the PureMessage milter is otherwise unavailable. If this value is not set to “T”, messages will bypass PureMessage when the milter is down or busy. See “Configuring Sendmail” in *Installing PureMessage for UNIX* for more information.

Is the PureMessage Policy configured to deliver messages before they are scanned for spam?

Delivery actions can be assigned to any policy rule, regardless of the rule's position in the policy script. If a message matches a rule with a delivery action before being processed by a rule that scans for spam, the message is delivered without scanning.

Related concepts

[Configuring an External Sendmail Installation](#) (page 45)

Related information

[pmx.conf](#)

3.9.2 Why didn't PureMessage quarantine a message that is spam?

There are a variety of reasons why a message containing spam characteristics is not identified as spam and treated accordingly. This may be due to aspects of PureMessage configuration described below.

Anti-Spam Engine

Ensure that PureMessage is using the latest anti-spam engine package by navigating to the **Support > Check for Updates** page in the PureMessage Manager, clicking **Query**, and checking that there is no update available for the **PureMessage-AntiSpam-Engine**. If there is, run `pmx-setup` at the command line to launch the installer and retrieve available update(s).

Anti-Spam Data

Ensure that PureMessage is using the latest anti-spam data package by navigating to the **Support > View Installed Packages** page in the PureMessage Manager, and examining the date of the **PureMessage-AntiSpam-Data** package. It should be the current day's date. If it isn't, check the **Support > Check for Updates** page, as described above, and update the package by running `pmx-setup` at the command line.

Anti-Spam Opt-Outs

If the recipient's address is included in the Anti-spam opt-outs list or the sender's address is included in the Whitelisted senders list, the message is exempt from anti-spam filtering. See “Editing Lists” in the *Manager Reference* for more

Trusted Relay Configuration

information. Also check that **MTA IP Blocking** is enabled.

PureMessage includes the ability to specify the IP addresses of external relays that are known to be “safe”. Ensure that trusted relays are configured and enabled. See the **Policy > Trusted Relay IPs** section of the *Manager Reference* for instructions.

Network DNS Access

A number of spam detection techniques rely on access to DNS servers. If DNS-based network checks are enabled (the default), ensure that the DNS server is functioning properly and communicating with the server(s) where PureMessage is running.

Quarantine Threshold in Policy Script

The PureMessage policy script performs actions on messages based on their spam probability. For example, the policy script can be configured to quarantine messages if they have a spam probability of 50% or greater. Changing probability-based actions in the policy script (via the `pmx-policy` command-line program or via the **Policy** tab in the PureMessage Manager) can possibly result in some spam not being detected.

Email Headers

If the message is subject to filtering but PureMessage has not identified it as spam, examine the message to see what headers were added by PureMessage during processing. By default, the `X-PMX-Version` header is added to all messages from external hosts. The absence of this header indicates that PureMessage has not processed the message. The default policy script also adds an `X-PerlMx-Spam` header to all messages with a spam probability. If the message's spam probability exceeds 50%, PureMessage not only adds the `<X-PerlMx-Spam>` header, but also alters the subject line and copies the message to the quarantine. The presence of this header indicates that anti-spam processing was completed. See “Policy Configuration” in the *Administrator's Reference* for more information.

message_log

If the message does not have an `X-PerlMx-Spam` header, you can check the `message_log` (by default, `/opt/pmx6/var/log/message_log`) to see what spam score the message received. The log file can be analyzed to determine the message's interaction with the policy script.

Note

You can help Sophos in its continuing efforts to improve the accuracy of PureMessage spam *heuristics* by forwarding misidentified items as attachments to:

- is-spam@labs.sophos.com for spam messages that escaped detection
- not-spam@labs.sophos.com for messages that were incorrectly identified as spam

You can also share your aggregated message statistics with Sophos by ensuring that **Support > Share data with Sophos** is enabled.

Related concepts

[Policy Configuration](#) (page 242)

This section describes the customization and fine tuning of message filtering for viruses, spam, and organizational policy compliance.

[Configuring Spam Detection](#) (page 275)

Related tasks

[Editing Lists](#) (page 120)

Related information

[re.rules](#)

[pmx-policy](#)

3.9.3 Why has the spam catch rate suddenly declined?

If the general effectiveness of spam detection declines, it is most likely related to the following issues:

Ensure that PureMessage is using the latest anti-spam engine package by navigating to the **Support > Check for Updates** page in the PureMessage Manager, clicking **Query** to see if an update is available for the **PureMessage-AntiSpam-Engine**. If there is, run `pmx-setup` at the command line to install the available update(s).

Ensure that PureMessage is using the latest anti-spam data package by navigating to the **Support > View Installed Packages** page in the PureMessage Manager, and examining the date of the **PureMessage-AntiSpam-Data** package. It should be the current day's date. If it isn't, check the **Support > Check for Updates** page, as described above, and update the package.

The issues described above affect overall spam detection. To troubleshoot a specific message, see "Why didn't PureMessage quarantine a message that is spam?" Also, see "Deployment Strategies" in the *Getting Started Guide* for information about configuring message-specific aspects of spam detection, such as whitelists, blacklists and custom anti-spam rules.

Related concepts

[Deployment Strategies](#) (page 6)

[PureMessage Default Lists, see Trusted Relay IPs](#) (page 118)

[Why didn't PureMessage quarantine a message that is spam?](#) (page 305)

Related tasks

[Configuring Anti-Spam Options](#) (page 127)

Related information

[pmx-policy](#)

3.9.4 What happens to messages sent to is-spam and not-spam?

If messages that should have been identified as spam are not caught by PureMessage filtering, you are encouraged to forward these messages to is-spam@labs.sophos.com.

Conversely, if PureMessage has filtered messages that you do not consider to be spam, you should forward them to not-spam@labs.sophos.com. Forwarded messages are processed as follows:

- The message is received and checked for attachments. It is preferred that you forward spam messages as “message/rfc822” attachments. Submissions received without one or more spam samples are discarded.
- Spam samples forwarded as “message/rfc822” attachments are automatically extracted and sent to our spam database. URLs, IP addresses, phone numbers, content signatures and other message attributes are automatically extracted from the spam messages and passed to an automated assessment system.
- An automated assessment is made as to whether the information extracted from the submitted spam e-mail will enable PureMessage to block the message. If PureMessage cannot consistently filter the spam message using the extracted information, the information is passed along to Sophos Labs analysts for manual analysis.
- The anti-spam data that is extracted or developed through this process is immediately added to the latest anti-spam package, which is made available via an automatic update.

3.9.5 Troubleshooting Quarantine Digests

This section provides information to assist with troubleshooting common problems encountered when generating quarantine digests.

Note

The commands described in the following procedures may vary, depending on your operating system, mail transfer agent (MTA), and PureMessage installation parameters (such as the installation path or quarantine location). Alter as necessary. All commands should be run as the PureMessage user ('pmx6' by default).

Digests Are Not Generated

1. View the list of Background Services in the PureMessage Manager, and ensure that the Queue Runner service is running.
2. Verify that the following command, generates digests and provides verbose feedback, is running as a scheduled job in the PureMessage Manager. As the PureMessage user ('pmx6' by default), run this command, and watch for errors:

```
pmx-qdigest -verbose
```

3. Run the appropriate commands to verify that PureMessage and your MTA are running:

Verify PureMessage Status (All Platforms)

```
pmx-status
```

Verify MTA Status (Postfix)

On Linux or BSD, run: `ps -uwx|grep postfix`

On Solaris, run: `ps -ef|grep postfix`

Postfix should be displayed as a running process.

Verify MTA Status (sendmail)

On Linux or BSD, run: `ps -uwx|grep sendmail`

On Solaris, run: `ps -ef|grep sendmail`

Sendmail should be displayed as a running process.

4. Verify that PureMessage and your MTA are interacting by running the following command and watching for message activity.

```
pmx-mlog -verbose
```

This command reads the PureMessage message log. If there is no message activity, proceed to the next step.

5. Verify that a line similar to one of the following exists:

Postfix

In the `/opt/pmx6/postfix/etc/main.cf` configuration file, search for:

```
content_filter = pmx:127.0.0.1:10025
```

Sendmail

In the `/opt/pmx6/sendmail/etc/sendmail.mc` configuration file, search for:

```
INPUT_MAIL_FILTER
('Policy','s=inet:3365@localhost,F=T,T=C:5m;\
E:8m;R:4m;S:2m') dnl
```

If the line for your MTA exists, but there is still no message activity, verify that the correct MTA binary is being used (for example, the MTA version of Postfix included with PureMessage, and not another version of Postfix that is also installed on the system).

Some Users Are Not Receiving Digests

1. Ensure that the user in question is a member of the **Quarantine digest users** list. View the list members on the **Policy** tab of the PureMessage Manager.
2. Verify that there are messages in the quarantine for the user in question that have not previously been included on a digest.

To check which messages have been included in digests, enter the following command on the command line:

```
pmx-qdigest --dump
```

Enter `man pmx-qdigest` for information about sorting and filtering output from this command.

Users Can't Release Quarantined Messages

- When the user requests the release of a message from the quarantine by replying to a quarantine digest, verify that the message reaches the PureMessage server by checking the sendmail maillog. If digest replies from the end user are not reaching the PureMessage server, verify that the mail routing is correct.
- Check that the message being requested has not been deleted from the quarantine by the `pmx-qexpire` scheduled job. Compare the date of the message with the value of `expire_time` in `etc/pmx.d/quarantine_expire.conf`.
- Check that the quarantine digest itself has not expired by comparing the date of the digest with the value of `expire` in `/opt/pmx6/etc/pmx-qdigest.conf` (5 days by default).

Error Message: Can't call method min_digest_id on an undefined value

The following `pmx-qdigest` error may be caused by a corrupted quarantine scan database (`/opt/pmx6/var/qdigest/scan.db`).

```
Can't call method "min_digest_id" on an undefined value /opt/pmx6/bin/
pmx-qdigest line n
```

- Verify this with the following command:

```
pmx-qdigest --dump
```

If it does not return a list of quarantine IDs for each digest type and user, and instead returns an error like the one above, the `scan.db` file is almost certainly the cause.

- Check if the permissions and ownership of the file are correct:

```
permissions: -rw-r--r--
owner:      pmx
group:      pmx
```

- If these are incorrect, changing them with `chmod` or `chown` may solve the problem. If the permissions and ownership are correct, or if changing them back to the defaults does not solve the problem, generate a new `scan.db` file:

```
cd 'pmx prefix'/var/counters
mv scan.db scan.db.broken
pmx-qdigest --earliest <YYYY-MM-DD hh:mm:ss>
```

Set `<YYYY-MM-DD hh:mm:ss>` to the time of the earliest message to include in the digest. The quarantine scan will start from this point.

Once the `pmx-qdigest` command has completed, the `scan.db` database should be regenerated. The next quarantine digest will start with the last quarantine ID scanned during the run (in this case, the most recent message in the quarantine).

The next time `pmx-qdigest` is run by the Scheduler, it should complete normally.

3.9.6 Troubleshooting the End User Web Interface

Log Files

The log file locations listed below are relative to `/opt/pmx6/var/log`.

- `manager/eui_local_log`
This file logs transactions and critical errors that are generated by the End User Web Interface (EUWI).
- `manager/httpd_access_log`
The HTTPD service access log.
- `manager/httpd_error_log`
The HTTPD service error log.

How a Transaction Appears in the Logs

Each successful EUWI transaction logs to two files, in this order:

1. `eui_local_log` (on the EUWI server if installed separately).

Entries for EUWI page submissions look something like:

```
1 2005-03-17T14:23:09 [5646,eui_local] quarantine_qview user
  'user@example.com',\
  session '66705d63e810acd456f38831d0bd4ba9'
```

2. `httpd_access_log` (on the EUWI server if installed separately).

Entries for various EUWI transactions. These transactions should look something like:

```
192.168.99.135 - - [17/Mar/2005:14:23:09 -0800] "POST /index.cgi
HTTP/1.1" 302 7192
192.168.99.135 - - [17/Mar/2005:14:23:11 -0800] "GET /messages.cgi?
message_type=blocked\
HTTP/1.1" 200 7605
192.168.99.135 - - [17/Mar/2005:14:23:11 -0800] "GET /main.css
HTTP/1.1" 304 -
```

If errors are encountered by the HTTPD service, they are logged to `httpd_error_log`.

Authentication and transaction errors are logged to `eui_local_log`.

Other errors are logged to `error_log`.

Common Error Messages

A brief description of some common error messages.

Invalid user in session

The session cookie does not match the user, or the user attempted to log in using an old session password. An email address and session ID will be shown if available.

Temporarily unavailable

To resolve this, the user must log out and log in again with the most recent session password sent by the EUWI.

Your session has expired or is invalid. Please log in again

The `enduser.conf` enabled setting is set to a false value.

The session cookie or session ID is stale or invalid, or there was an error when communicating with the RPC back end. Check the `httpd_error.log` and `rpc_error.log` files for more specific error messages.

This error is also triggered when the user does not appear in, or no longer matches against the configured **enduser-users** list.

Note

If either the `enduser-users` file or the `rpc-hosts` file is missing or invalid, the EUWI user will get an error like this. The real reason for the failure will, for security reasons, not be communicated to the end user, but will be visible to the admin in the log file.

`user_invalid_access_list="The configured end user access list '<name>' is invalid and cannot be used."`

The `rpc.conf` file contains an invalid list or the contents of that list are invalid.

Authentication failed for user ...

Indicates a problem with the configured authentication back end that prevents the user from logging in. This can be caused when the username does not match against the `enduser-users` list, or by a bad password, a bad username, or an invalid session. The errors in the log files provide more detail.

If the cause of the error is still not clear, increase the `debug_level` in `enduser_ui.conf`, restart the HTTPD (RPC/UI) service and attempt the action again. The logs should contain a more detailed error message.

RPC call failed: ...

In general, an error message in this format is a transport error. There was some condition preventing the central server from supplying a more appropriate error, either because it could not be contacted or because a library used by the back end died at an unexpected time.

Unable to initialize database connection:...

Indicates that the database is not running, or that it is not possible for the EUWI to establish a connection to the database. Check that the database has been started. In the case of a multi-server install, check `postgres/var/data/pg_hba.conf` to ensure that the EUWI server

has been granted permission to access the database.

4 Policy Script Tutorial

This tutorial describes the syntax used in the policy script, analyzes the default PureMessage policy script, and shows examples of common policy script modifications.

PureMessage filters email according to the configuration contained in the policy script (`policy.siv`, located by default in the directory `/opt/pmx6/etc`). The policy script can be modified via the PureMessage Manager, or can be edited with standard text editing programs such as `vi`.

Refer to the `pmx-policy` manpage for the policy command reference. Refer to “Policy Configuration” in the *PureMessage User Guide* for a general overview of the PureMessage policy, and links to the PureMessage Manager interface for modifying policies.

For information about specific PureMessage policy tests and actions, see the `pmx-policy` man page.

Sieve is a language used for filtering email messages. It is a multi-vendor effort, and has been proposed as a standard to the Internet Engineering Task Force. PureMessage makes use of Sieve for filtering email via policies. Filter parameters are stored in the file `policy.siv`, located by default in the `/opt/pmx6/etc` directory. For information about manually modifying the Sieve code in the `policy.siv` file, see the `pmx-policy` man page and the web pages http://www.gnu.org/software/mailutils/manual/html_node/mailutils_93.html and <http://www.ietf.org/rfc/rfc3028.txt>

Related concepts

[Policy Configuration](#) (page 242)

This section describes the customization and fine tuning of message filtering for viruses, spam, and organizational policy compliance.

Related information

[pmx-policy](#)

4.1 Policy Script Syntax Overview

This section introduces the PureMessage policy syntax. Examples in this overview use pseudocode to best describe policy syntax structure and generically demonstrate various PureMessage mail filtering scenarios. Specific PureMessage policy syntax is discussed later in PureMessage Default Policy Overview.

The PureMessage policy script, called `policy.siv`, is written using a variation of the Sieve mail filtering language. Sieve is a simple command-based scripting language; its syntax resembles C or Perl, but it has no variables or looping constructs. In its simplest form, a Sieve script is a set of commands executed in a sequence. The commands in the PureMessage `policy.siv` script can be actions, action modifiers, tests, controls, blocks or comments.

The following example identifies the commands in a policy filter:

```
# This script performs a virus test.
# Messages containing viruses are filed as SPAM and then quarantined.
if pmx_virus {
    pmx_file "SPAM";
    pmx_quarantine "Virus";
    stop;
}
```

Description:

- The '#' characters are Sieve comments.
- The `if` and `stop` commands are controls.
- The `pmx_virus` command is one example of many possible tests in PureMessage.
- The '{' and '}' characters delimit blocks.
- The `pmx_quarantine` and `pmx_file` commands are PureMessage action
- The "Virus" and "SPAM" strings are action modifiers.
- Actions are delimited with the ';' character.

Related concepts

[PureMessage Default Policy Overview](#) (page 323)

4.1.1 Actions

Policy actions can modify the routing or content of a message. Possible actions include: `keep;` (deliver message), `reject;` (reject message), and `pmx_quarantine;` (quarantine message). Actions always end with a semicolon. Semicolons delimit when an action ends.

Generic Example:

```
action;
```

Policy Script Example:

```
keep;
reject;
pmx_quarantine;
```

Action Modifiers

Action modifiers are optional parameters that alter the outcome of a particular policy action. For example, the `pmx_quarantine` action modifier is a string argument that specifies the reason that PureMessage quarantined a message. Reasons could include 'Virus', 'spam', or 'Attachment Over 100K'.

Generic Example:

```
action <action-modifier>;
```

Policy Script Example:

```
pmx_quarantine "Spam";
```

Note

Not all PureMessage action modifiers are of type 'string'. See the `pmx-policy` manpage for a definitive list of all syntax parameters and action modifiers.

Related information

[pmx-policy](#)

4.1.2 Blocks

PureMessage policy actions are grouped together in blocks. Blocks act like containers; they hold actions together. A block begins with an opening brace “{” and ends with a closing brace “}”. Every action in a block must end with a semicolon. The block itself ends with a closing brace.

Generic Example:

```
{  
    action;  
}
```

Policy Script Example:

```
{  
    keep;  
}
```

In this example, the policy block contains the single action `keep`. The `keep` command delivers the message to all envelope recipients. Blocks can, however, contain one or many actions. When multiple actions are listed inside a block, each action is executed in a sequence from top to bottom.

Policy Script Example:

```
{  
    pmx_quarantine "Blacklisted";  
    stop;  
}
```

In this example, the policy block contains a sequence of two mail filtering actions. The message is first quarantined with the reason “Blacklisted”, and then the script stops. Each action is delimited with a semicolon to indicate to the interpreter when each action ends.

4.1.3 Controls

Control commands determine whether a block of actions are executed, or if no further actions should execute. Control keywords are positioned with an associated test command before the opening brace in a block to help control the order of execution through the policy script. Note that tests are covered further in the next section of the tutorial. For now, understand that a test command evaluates to either “true” or “false”. The PureMessage `policy.siv` script uses three control keywords to filter messages: `if`, `else`, and `stop`.

Related concepts

[Tests](#) (page 318)

if

The most basic control command is the `if` keyword. When the interpreter sees an `if` command, it evaluates the associated test. If the test is “true”, the actions within the block execute.

Generic Example:

```
if test {
    action;
    next_action;
}
```

Policy Script Example:

```
if pmx_virus {
    reject "Virus Found in
Message";
    stop;
}
```

This example tests if a message contains a virus. If the test is “true” and the message contains a virus, the actions within the block execute from top to bottom. The message is rejected with reason “Virus Found in Message”, and message filtering stops. But what happens if the test is “false” and the message is legitimate?

else

The `else` command can execute when an `if` command fails. The `if` and `else` control statements work together in a policy script to filter messages.

Generic Example:

```
if test {
    action;
    next_action;
}

else {
    action;
    next_action;
}
```

In this example, if the test is “true”, execute the first set of statements. `else`, execute the second.

Policy Script Example:

```
if pmx_virus {
    reject "Virus Found in
Message";
    stop;
}

else {
    keep;
}
```

In this policy filter, the `if` and `else` control statements work together with tests and actions to either reject a virus-laden message or to deliver virus-free mail to envelope recipients. The `else` command executes when the `if` statement fails.

stop

The `stop` control halts execution of the policy script. Unlike the `if` and `else` control commands, the `stop` command is used within a block to end message filtering when the required actions have executed and message filtering is no longer required.

Generic Example:

```
if test {
    action;
    next_action;
    stop;
}
```

Policy Script Example:

```
if pmx_relay :memberof
"whitelisted-hosts" {
    keep;
    stop;
}
```

In this example, the filter delivers the message if the host is on the PureMessage “whitelisted-hosts” list. The `stop` control command ends the policy script after message delivery.

4.1.4 Tests

Tests act as block gatekeepers. They check whether a certain condition in the policy script is met. A test is positioned with a control command before the opening brace in a block to determine whether the policy actions should execute. If a condition is met, the test is “true” and the actions within the block execute. If the test fails, the condition is “false” and no actions will execute for that particular policy.

Generic Example:

```
if test {
    action;
    next_action;
}
```

Policy Script Example:

```
if pmx_virus {
    reject "Virus Found in
Message";
    stop;
}
```

In this example, a message must pass the `pmx-virus` test before the actions in the block can execute. If the test is “true” and the message contains a virus, the message is rejected and the policy script stops filtering the message. If the message does not contain a virus, the test fails and no actions from the block will execute.

Match-Types

A match-type performs a matching operation on a particular feature of a message. In the policy script, the match-type modifies a test to check for specific characteristics.

For example, the `:is` match-type is used here to modify the envelope test to determine if a message is "From" "spammer@foo.com".

```
if envelope :is "From" "spammer@foo.com" {
    pmx_quarantine "SPAM";
}
```

PureMessage uses several match-types to filter messages. These include: `:memberof` for matching content in lists, `:contains` and `:is` for matching strings, `:over` or `:under` for numeric comparison, and `:re` for searching messages with regular expressions. Depending on the specific match-type used, additional parameters will have to be specified. See "MATCH-TYPE" in the `pmx-policy` manpage guide for more information.

Generic Example:

```
if test :match_type {
    action;
    next_action;
}
```

Related concepts

[Lists](#) (page 319)

[Strings](#) (page 320)

[Numeric Comparison](#) (page 320)

[Regular Expressions](#) (page 320)

Related information

[pmx-policy](#)

Lists

Use the `:memberof` match-type tag to determine if a sender or host is on a particular PureMessage list. Lists are enclosed in quotes when passed as arguments with the `:memberof` match-type tag.

Policy Script Example:

```
if envelope :memberof "From"
"whitelisted-senders" {
    keep;
    stop;
}
```

In this example, the match-type `:memberof` takes a PureMessage "whitelisted-senders" list as an argument. If the message sender is matched with a sender on this PureMessage list, the message is delivered to the envelope recipient.

Strings

Use the `:contains` or `:is` match-type to determine if a text string can be matched against content in a message. Strings are enclosed in quotes when passed as arguments with the `:memberof` and `:is` match-type tags.

Policy Script Example:

```
if header :contains "Buy Now!" {
    pmx_quarantine "SPAM";
    stop;
}
```

In this example, the match-type `:contains` takes a text string as an argument. If the string “Buy Now!” is matched with content in any message header, the message is quarantined with the reason “SPAM”. The `stop` command ends filter execution.

Numeric Comparison

Use either the `:over` or `:under` match-type tags to specify a numeric comparison. Both tags take a number as an argument to determine whether a test is “true” or “false”. For example, the `:over` tag can compare a message’s spam probability to determine if a particular message should be delivered.

Policy Script Example:

```
if pmx_spam_prob :over 50 {
    pmx_quarantine;
    stop;
}
```

In this example, the `:over` match-type is used to perform a numeric comparison. The `:over` match-type takes a number argument. The test is “true” if the message has a spam probability of over 50%.

Policy Script Example:

```
if size :over 100K {
    pmx_file;
    discard;
}
```

This example tests to see if the total message size is over 100K. If the message is over 100K, a copy of the message is filed in the quarantine. The message is then discarded.

Regular Expressions

Regular expressions can be used as match-type operators for many types of policy rule tests. For example, a regular expression can be used to test the contents of a message’s “Envelope To” field.

Certain policy rule actions also support the use of regular expressions. For example, the “Deliver immediately for” action supports the use of a regular expression as a match-type for exceptions.

Policy Script Example:

```
if pmx_relay :re ".*\.com$" {
    keep;
    stop;
}
```

In this example, the match-type `:re` is used to perform a regular expression comparison. If the sender's address ends in the string `".com"`, the message is delivered and the filter stops evaluating the message.

Note

The `:re` match-type is a PureMessage extension to the Sieve language. When regular expressions are used in PureMessage policy rule tests or actions, they are not prefixed or suffixed with slashes or braces. However, when manually editing the policy script on the command-line, you must “escape” backslashes, quotes and periods within regular expressions by preceding them with a backslash. For example, to search for the string `"and\or"`, enter `"and\\or"` as the regular expression. The PureMessage Manager, a web-based interface for managing PureMessage, will automatically escape these characters for you.

For information about using regular expressions with the policy script via the PureMessage Manager, see “Policy Rule Tests” and “Policy Rule Actions” in the Policy section of the PureMessage Administrator's Reference. If you are manually editing the policy script, see the `pmx-policy` manpage (the documentation for the command-line interface to the PureMessage policy engine). See the “Regular Expressions Primer” in the PureMessage *Administrator's Reference* for general information about regular expressions.

Related information

[pmx-policy](#)

Compound Tests, `allof` and `anyof`

Some tests in the PureMessage policy script are used as logical operators. These tests are called compound tests as they take other tests as arguments. Use the `allof` or `anyof` commands to build a compound test. The `allof` command functions as logical AND. (All compound tests in the `allof` list must be “true” for the corresponding actions in a block to execute.) The `anyof` command functions as a logical OR. (Only one of the compound tests in the `anyof` list needs to be “true” for the actions in the block to execute).

In the policy script, the `allof` and `anyof` commands take a test and a list as arguments. To build a compound test, group individual tests and associated arguments in parentheses before the beginning of a block statement. Each compound test in the list must be separated with a single comma “,” to delimit the end of each test.

Generic Example:

```
if anyof (test :match_type "list",
    test :match_type "list") {
    action;
    next_action;
}
```

In this example, the `anyof` command takes two tests. Each test takes a match-type with a list argument. The compound test is enclosed in parentheses. Each test is separated with a comma character.

Policy Script Example:

```
if anyof (host :memberof
"whitelisted-hosts",
        sender :memberof
"whitelisted-senders") {
    keep;
    stop;
}
```

This policy filter evaluates both the host and sender for membership on a PureMessage “whitelist”. If either the host or sender are found on a “whitelist”, the message is delivered to all envelope recipients. The script then halts with the `stop` command.

4.1.5 Comments

Comments are used in the policy script to document how a particular test, action, or control statement contributes to a filter. For each line of commented text, use the “#” character followed by a single space to denote information the Sieve interpreter should ignore.

Note

Comments should not be placed within multi-line strings.

Generic Example:

```
# Comments
# More comments
if test { # This is a valid
comment!
    action;
    next_action;
}
```

Policy Script Example:

```
# This filter scans all message
headers for the words "Buy Now!"
# If this string is
found, PureMessage rejects the
message
# and the filter stops evaluating
the message for spam.
if header :contains "Buy Now!" {
    reject;
    stop;
}
```

4.2 PureMessage Default Policy Overview

This section reviews the structure, syntax, and execution of each filter in the default `policy.siv` script. The policy script is executed for each message processed by PureMessage. It can test various characteristics of the message, and perform a variety of actions based on the results of those tests.

Note

The default policy script is installed and enabled during PureMessage installation. The default policy varies according to your PureMessage license; for example, if you do not have a license for the PureMessage Virus component, virus-checking rules are not configured.

4.2.1 PureMessage Sieve Extensions

The PureMessage `policy.siv` script uses extensions to the Sieve language to filter mail. Each PureMessage specific extension is prefixed with `pmx_` to identify the action, test, or command as unique to the program. To use these extensions include the command `"require PureMessage;"` at the beginning of the policy script. PureMessage also provides the standard extensions `reject` and `envelope` as specified in RFC 3028. Note that both `reject` and `envelope` are included implicitly with `require "PureMessage";`. See the `pmx-policy` manpage for more on specific PureMessage Sieve extensions.

Lastly, PureMessage has an extension for using regular expressions in match-type tests. See Regular Expressions.

Related concepts

[Regular Expressions](#) (page 320)

Related information

<http://www.ietf.org/rfc/rfc3028.txt>

4.2.2 `policy.siv`

In general, the default `policy.siv` file is comprised of eight `if` control statements and one `else` statement. Each control statement is comprised of a type of test, a block and an associated sequence of actions. Action sequences can either modify or route particular messages in PureMessage. Actions, tests, and commands specific to PureMessage are prefixed with `pmx_` to identify the command as an extension to Sieve.

For the purpose of discussion, the `policy.siv` file is broken into six parts. Each script piece is followed by a detailed description of the commands and syntax used to filter mail in PureMessage.

Policy Script 1: Scan and Deliver Internal Messages

```
pmx_test_mark;
# attr NAME=Mail from internal hosts
if pmx_relay :memberof "internal-hosts" {
    # The 'pmx-mlog-watch' depends on this to know which messages
    # are outgoing and which are not.
    pmx_mark1 "i";
    # attr NAME=Reject mail containing viruses
    if pmx_virus {
        reject "One or more viruses were detected in the message.";
        stop;
    }
}
```

Description:

- The `pmx_test_mark` command is used to recognize sample messages sent by the “`pmx-test`” program.

Note

Consider disabling this action if running a site with a high-mail volume as this command will delay relay tests. See the `pmx-policy` manpage for further details.

- If the `pmx_relay` test finds the message originated from a relay defined in the “internal-hosts” list:
 - The `pmx_mark1` command adds a “i” mark to the message log to enable “Perimeter Protection” to distinguish outgoing messages from internal hosts. Messages accumulate “marks” throughout the filtering process. These marks can be used later to generate custom statistical reports.
- The `pmx_virus` command scans the message for virus threats. If the message contains a virus:
 - The message is rejected with the reason “One or more viruses were detected in the message”.
 - The stop control ends message processing.

Related information

[pmx-policy](#)

Policy Script 2: Scan External Mail for Viruses

```
# attr NAME=Mail from external hosts
else {
    pmx_add_header "X-PMX-Version" "%%PMX_VERSION%%";
    pmx_mark "Size" "%%MESSAGE_SIZE%%";
    # attr NAME=Clean mail containing viruses
    if pmx_virus {
        pmx_file "Virus";
        pmx_virus_clean "cantclean.tmpl";
        pmx_replace_header "Subject" "[PMX:VIRUS] %%SUBJECT%%";
        pmx_replace_header "X-PerlMx-Virus-Detected" "%%VIRUS_IDS%%";
    }
}
```

Description:

- If the `if pmx-relay` test is “false”, and the else control is executed:
 - Mail is assumed to be from an external host.
 - The `pmx_add_header` command adds a header to the message. The header includes the string “X-PMX-Version” and the PureMessage version (which is added using the `%%PMX_VERSION%%` template variable.)
 - The size of the message is then written to the message log. The `%%MESSAGE_SIZE%%` template variable is substituted for the actual size of the message, in bytes.
- The `pmx_virus` command tests the message for virus threats. If the test is “true”, and the message contains a virus:
 - `pmx_file` copies the message to the quarantine with the reason “Virus”. This action does not affect the delivery of the message.
 - The `pmx_virus_clean` action attempts to clean the virus from the message. If cleaning fails, the message is quarantined, and a message is sent to the recipient based on the specified failure template file, `cantclean.tmpl`.
 - The `pmx_replace_header` command prefixes the “Subject” header with “ [PMX:VIRUS] ”. The original “Subject” is added to the end of the header with the `%%SUBJECT%%` template variable.
 - The `pmx_replace_header` command then adds, or alters, a “X-PerlMx-Virus-Detected” header. The `%%VIRUS_IDS%%` template variable adds the found virus IDs to the message header.

Policy Script 3: Deliver External Whitelisted Messages

```
# attr NAME=Deliver mail from whitelisted hosts and senders
if anyof(pmx_relay :memberof "whitelisted-hosts",
        envelope :memberof "From" "whitelisted-senders") {
    keep;
    stop;
}
```

Description:

- If the message originates from a relay defined in the “whitelisted-hosts” list OR the message has an “Envelope From” address defined in the “whitelisted-senders” list:
 - The `keep` action accepts the message for delivery.
 - The `stop` control ends message processing.

Policy Script 4: Deliver External Messages to Anti-Spam Opt-Outs

```
# attr NAME=Deliver mail to anti-spam opt-outs
if true {
    pmx_deliver_for :memberof "anti-spam-optouts";
}
```

- If the message recipient address is defined in the “Anti-Spam Opt-Outs” list:
 - The `pmx_deliver_for` “opt-in/opt-out” command delivers the message. This command matches against the message envelope recipient and opts them out of further filter processing.

Policy Script 5: Quarantine External Blacklisted Messages

```
# attr NAME=Quarantine mail from blacklisted hosts and senders
if anyof(pmx_relay :memberof "blacklisted-hosts",
         envelope :memberof "From" "blacklisted-senders") {
    pmx_quarantine "Blacklisted";
    stop;
}
```

Description:

- If the message originates from a relay defined in the “blacklisted-hosts” list, or the message has an “Envelope From” address defined in the “blacklisted-senders” list:
 - The `pmx_quarantine` action sends the message to the PureMessage quarantine with the reason “Blacklisted”.
 - The `stop` control ends message processing.

Policy Script 6: Calculate Spam Probability, Modify and Deliver

Modify Message

```
# import levels here
# attr NAME=Copy to quarantine and deliver if spam probability is 50%
# or more.
if not pmx_spam_prob :under 50 {
    pmx_replace_header "X-PerlMx-Spam" "Gauge=%%XGAUGE%%%IGAUGE%%,
        Probability=%%PROB%%, Report='%%HITS%%'";
    pmx_file "Spam";
    pmx_replace_header "Subject" "[PMX:%%GAUGE%%] %%SUBJECT%%";
    stop;
}
```

Description:

- If the `pmx_spam_prob` test finds the message has a spam probability of 50% or more:
 - `pmx_replace_header` adds, or alters, an “X-PerlMx-Spam” header. The `%%XGAUGE%%` `%IGAUGE%` variables add an “X” symbol to the header for every 10% of spam probability identified through the `%%PROB%%` variable. The `%%HITS%%` variable then adds a list of spam features found by the engine.
 - The `pmx_file` command then copies the message to the quarantine with the reason “Spam”.
 - The `pmx_replace_header` command prefixes the message “Subject” header with “PMX:” and a “#” symbol for every 10% the message’s spam probability exceeds 50%. At least one ‘#’ is always appended to each message, indicating that the message contains 0-50% spam. The original message subject is added to the end of the header with the `%%SUBJECT%%` template variable.
 - The `stop` control ends message processing.

Deliver Message

```
# attr NAME=Add X-Header and deliver messages
else {
    pmx_replace_header "X-PerlMx-Spam" "Gauge=%%XGAUGE%%%IGAUGE%%,
        Probability=%%PROB%%, Report='%%HITS%%'";
    stop;
}
```

Description:

- If the `pmx_spam_prob` test finds the message has a spam probability of less than 50%:
 - `pmx_replace_header` adds, or alters, an “X-PerlMx-Spam” header. The `%%XGAUGE%%` `%IGAUGE%` variables add an “X” symbol to the header for every 10% of spam probability identified through the `%%PROB%%` variable. The `%%HITS%%` variable then adds a list of spam features found by the engine.

- The `stop` control then ends message processing.

4.3 Customizing the PureMessage Default Policy

This section describes how to customize the PureMessage default policy to build mail filters suitable for your environment. Various filtering scenarios are provided to illustrate different approaches to policy modification. These scenarios are:

4.3.1 Adding a Recipient

Occasionally, a site may require certain messages to be automatically sent to a known address. For example, all messages containing the keyword “bug” might be sent to an bug tracking system. This policy adds a new recipient to a message (`bugs@example.com`) when the “Subject” header has the string “bug” in it, the recipient is “old-bugs@example.com”, or the header “X-Bug-Id” exists.

```
# attr NAME=add a recipient
# Adds a "bugs@example.com" recipient to the envelope.
if anyof (header :matches "Subject" "*bug*",
           envelope "to" "old-bugs@example.com",
           header :matches "X-Bug-Id" "*") {
    pmx_add_recipient "bugs@example.com";
}
```

Description:

- The first command is an `if anyof` statement comprised of three compound tests. The `header` test returns “true” if the “Subject” header matches the string “bug”. The `envelope` test returns “true” if `anyof` the message’s recipient “To” headers match the string “old-bugs@example.com”. The second header test returns “true” if the “X-Bug-Id” header matches itself. If any of these three tests return “true”:
 - The `pmx_add_recipient` command delivers the message to `bugs@example.com`. The message is also delivered to all original envelope recipients.

Where does this filter go?

The “add a recipient” filter should be placed before the filter in “Policy Script 1: Scan and Deliver Internal Messages” and directly after the `pmx_test_mark` action. Positioning the “add a recipient” filter at this point ensures that:

- The “add a recipient” filter always executes. (A `stop` command from another filter will not prevent it from executing.)
- Subsequent spam and virus filters will always execute. (The “add recipient” filter does not contain a `stop` command that would otherwise end the policy script before spam and virus filters execute.)

4.3.2 Adding a Header

Adding a new header to all messages can be useful for tracking purposes. This filter adds an “X-Seen-By” header to all messages.

```
# attr NAME=add a header
# Adds an 'X-Seen-By' header to all messages.
pmx_add_header "X-Seen-By" "%HOSTNAME%";
```

Description:

- The `pmx_add_header` action adds an “X-Seen-By” header to the message. The value of the header is the hostname of the PureMessage machine (for example `mail.example.com`). The `%HOSTNAME%` template variable is called. For a list of supported template variables, see the `pmx-policy` manpage.

Where does this filter go?

The “add a header” filter should be placed before the filter in “Policy Script 1: Scan and Deliver Internal Messages” and directly after the `pmx_test_mark` action. Positioning the “add a header” filter at this point ensures that:

- The “add a header” filter always executes. (A `stop` command from another filter will not prevent it from executing.)
- Subsequent spam and virus filters will always execute. (The “add a header” filter does not contain a `stop` command that would otherwise end the policy script before spam and virus filters execute.)

Related concepts

[Policy Script 1: Scan and Deliver Internal Messages](#) (page 324)

Related information

[pmx-policy](#)

4.3.3 Detecting Spam

In some instances, it may be preferable to deliver all spam messages to envelope recipients. For example, site administrators would consider this action when initially testing and optimizing PureMessage. In this situation, a spam detection filter is beneficial for identifying spam and delivering all messages to recipients with an associated spam probability. During the optimization process, mail recipients could then comment on the accuracy of a particular filter.

Note

The optional “PureMessage-Policy-Spam” component is required to use the “spam detection” filter.

```
# attr NAME=spam detection
# Detects spam probability over 50%.
# Prefixes 'subject' header with '[SPAM:]'.
# Adds an 'X-PMX-Spam' header.
if pmx_spam_prob :over 50 {
    pmx_replace_header "Subject" "[SPAM:%%GAUGE%%] %%SUBJECT%%";
    pmx_add_header "X-PMX-Spam" "Probability=%%PROB%%";
    stop;
}
```

Description:

- If the `pmx_spam_prob` test finds the message has a spam probability of 50% or more:
 - The `pmx_replace_header` action prefixes the “Subject” of the message with a string similar to: “ [SPAM:###] ”, where each additional “#” character denotes 10% above the argument to `pmx_spam_prob` (at least one “#” is always appended, indicating that the message contains 0-50% spam).
 Example: Using this filter, a message with a spam probability of 60% would have its “Subject” header prefixed with “ [SPAM:##] ”. A message with only 50% probability would have its “Subject” header prefixed with “ [SPAM:#] ”.
 - The `pmx_add_header` action then adds an “X-PMX-Spam” header to indicate the message’s numerical spam probability. The message header displays as follows:

```
X-PMX-Spam: Probability=63%
```

- The `stop` command then ends message processing.

Where does this filter go?

The “spam detection” filter replaces the “Copy to quarantine and deliver if spam probability is 50% or more” filter in Policy Script 6: Calculate Spam Probability, Modify and Deliver. Replacing the first part of this default policy filter with the “spam detection” filter ensures that:

- The “spam detection” filter executes when messages from external hosts contain over 50% spam.
- Messages with a spam probability of 50% or more are not quarantined. Headers are added with an associated spam probability.

Related concepts

[Policy Script 6: Calculate Spam Probability, Modify and Deliver](#) (page 327)

4.3.4 Quarantining Spam Messages

Use the following “quarantine spam” filter to decrease the number of spam messages directed to a recipient’s mailbox. The “quarantine spam” filter is a modification to the default policy script filter found in “Policy Script 6: Calculate Spam Probability, Modify and Deliver”. Unlike the default policy script filter,

the “quarantine spam” filter quarantines messages when the spam probability is over 80%. (The default policy filter only “files” spam messages when the spam probability is over 50%. See the `pmx_file` command in the `pmx-policy` manpage.

```
# attr NAME=quarantine spam
# Quarantines spam if probability over 80%.
# Else, delivers message if probability over 50%.
# Prefixes "Subject" header with "[SPAM:]".
# Adds an "X-PMX-Spam" header.
if pmx_spam_prob :over 80 {
    pmx_quarantine "Spam";
    stop;
}
elseif pmx_spam_prob :over 50 {
    pmx_replace_header "Subject" "[SPAM:%%GAUGE%%] %%SUBJECT%%";
    pmx_add_header "X-PMX-Spam" "Probability=%%PROB%%";
}
```

Description:

- If the `pmx_spam_prob` test finds the message has a spam probability of 80% or more:
 - The `pmx_quarantine` action copies the message to the PureMessage quarantine with the reason “Spam”. This string should be a single word. If multiple words are used the spaces between the words are silently changed to underscores. For example, the string “message is spam” becomes “message_is_spam”. Quarantined messages can be viewed, released and digested using the PureMessage Manager and the quarantine tools.
 - The `stop` command then ends message processing.
- If the `pmx_spam_prob` test finds the message has a spam probability of 50% or more:
 - The `pmx_replace_header` action prefixes the “Subject” of the message with a string similar to: “ [SPAM:###] ”, where each additional “#” character denotes 10% above the argument to `pmx_spam_prob` (at least one “#” is always appended, indicating that the message contains 0-50% spam).
 - The `pmx_add_header` action then adds an “X-PMX-Spam” header to indicate the message’s complete spam probability.

Where does this filter go?

The “quarantine spam” filter replaces the filter found in “Policy Script 6: Calculate Spam Probability, Modify and Deliver”. Replacing this default policy filter with the “quarantine spam” filter ensures that:

- The “quarantine spam” filter executes when messages from external hosts have spam probabilities of either over 80% or over 50%.
- Messages with a spam probability of 80% or more are quarantined.
- Messages are delivered, with additional headers, to envelope recipients if the spam probability is between 50% and 79%.

Related concepts

[Policy Script 6: Calculate Spam Probability, Modify and Deliver](#) (page 327)

Related information

[pmx-policy](#)

4.3.5 Catching Viruses

Use the following three “virus” filters to modify how PureMessage handles virus-laden messages.

Note

The optional “PureMessage-Policy-Virus” package is required when using any of the following virus filters.

Example 1: Quarantine all external messages containing virus variants.

Use the “virus 1” filter to quarantine all external messages containing virus variants. The “virus 1” filter is a modification to the default policy filter found in Policy Script 2: Scan External Mail for Viruses. Unlike the default filter, the “virus 1” filter quarantines all messages containing virus variants. No attempt is made to clean infected messages. See the `pmx_virus` command in the `pmx-policy` manpage.

```
# attr NAME=virus 1
# Quarantines all infected messages.
if pmx_virus {
  pmx_quarantine "Virus";
}
```

Description:

- If the `pmx_virus` test detects a virus in the message:
 - The `pmx_quarantine` action sends the message to the PureMessage quarantine with the reason “Virus”.

Where does this filter go?

The “virus 1” filter replaces the Policy Script 2: Scan External Mail for Viruses filter. Replacing this default policy filter with the “virus 1” filter ensures that:

- The “virus 1” filter executes when messages from external hosts containing virus variants are detected.
- All external messages containing virus variants are quarantined.
- Subsequent PureMessage policy filters will always execute. (The “virus 1” filter does not contain a `stop` command that would otherwise end the policy script before other PureMessage filters execute.)

Example 2: Attempt to clean all internal messages containing virus variants.

Use the “virus 2” filter to clean all internal messages containing virus variants. The “virus 2” filter is a modification to the default policy filter found in Policy Script 1: Scan and Deliver Internal Messages.

Unlike the default filter, the “virus 2” filter attempts to clean virus variants from all messages sent through internal hosts. The default policy rejects all internal mail containing viruses.

```
# attr NAME=virus 2
#
if pmx_virus {
    pmx_file "Virus";
    pmx_virus_clean "cantclean.tmpl";
    pmx_replace_header "Subject" "[PMX:VIRUS] %%SUBJECT%%";
    stop;
}
```

Description:

This filter attempts to clean the virus-laden message. If the message is successfully cleaned, it is sent to its original recipients. If the virus is not successfully cleaned, the infected part is replaced with the error template `cantclean.tmpl`. The “Subject” is marked with “ [PMX:VIRUS] ” to inform recipients that PureMessage found a virus.

Where does this filter go?

The “virus 2” filter replaces the Policy Script 1: Scan and Deliver Internal Messages filter. Replacing this default policy filter with the “virus 2” filter ensures that:

- The “virus 2” filter executes when messages from internal hosts containing virus variants are detected.
- All messages sent through internal hosts are cleaned if they contain virus variants.
- Subsequent PureMessage policy filters never execute. (The “virus 2” filter contains a `stop` command which ends the policy script and prevents other PureMessage filters from executing.)

Example 3: Discard external messages containing specific viruses.

Use the “virus 3” filter to evaluate mail sent through external hosts and to discard messages containing either the “Klez” or “Sobig” variants. The “virus 3” filter is a modification to the default policy filter found in Policy Script 2: Scan External Mail for Viruses. Unlike the default filter, the custom “virus 3” filter searches for specific viruses using the `pmx_virus_id` command.

```
# attr NAME=virus 3
# Discards messages infected with Klez or Sobig variants.
# Attempts to clean messages infected with other variants.
if pmx_virus {
    if pmx_virus_id :matches ["*Klez*", "*Sobig*"] {
        discard;
        stop;
    }
    pmx_file "Virus";
    pmx_virus_clean "cantclean.tmpl";
    pmx_replace_header "Subject" "[PMX:VIRUS] %%SUBJECT%%";
}
```

Description:

- The `pmx_virus` command tests the message for virus threats. If the test is “true”, and the message contains a virus:
 - The `pmx_virus_id` test checks if the message contains either the “Klez” or “Sobig” variants. If either virus is found:
 - The message is discarded.
 - The `stop` command ends message processing.
 - The `pmx_file` action then copies the message to the quarantine with the reason “Virus”.
 - The `pmx_virus_clean` action attempts to clean the virus from the message. If cleaning fails, the message is quarantined, and a message is sent to the recipient based on the specified failure template file, `cantclean.tmpl`.
 - The `pmx_replace_header` command prefixes the “Subject” header with `[PMX:VIRUS]`. The original “Subject” is added to the end of the header with the `%%SUBJECT%%` template variable.

Where does this filter go?

The “virus 3” filter replaces the “Policy Script 2: Scan External Mail for Viruses filter”. Replacing this default policy filter with the custom “virus 3” filter ensures that:

- The “virus 3” filter always executes when messages from external hosts containing specific virus variants are detected.
- All external messages containing specific virus variants are discarded.
- Subsequent PureMessage policy filters will always execute. (The “virus 3” filter does not contain a stop command that would otherwise end the policy script before other PureMessage filters execute.)

Related concepts

[Policy Script 1: Scan and Deliver Internal Messages](#) (page 324)

[Policy Script 2: Scan External Mail for Viruses](#) (page 325)

Related information

[pmx-policy](#)

4.3.6 Discarding Messages Based on Specific Characteristics

```
# attr NAME=discard
# Discards messages containing key phrases.
if header :matches "Subject" ["Re: Approved", "Re: Details", "Re: Movie",
                              "Re: My details", "Re: Thank you!"] {
    discard;
    stop;
}
```


Description:

- If the message “Subject” header matches any of the following strings in the list: “Re: Approved”, “Re: Details”, “Re: Movie”, “Re: My details”, “Re: Thank you!”
 - The message is discarded.
 - The `stop` command ends message processing.

Where does this filter go?

The “discard” filter should be added directly after the “Policy Script 5: Quarantine External Blacklisted Messages” policy filter. Positioning the “discard” filter at this point ensures that:

- The “discard” filter only executes after other PureMessage policies have filtered out spam, viruses, whitelists, blacklists, and opt-out lists.
(A `stop` command from another filter will not prevent it from executing.)
- Subsequent filters will never execute. (The “discard” filter uses a `stop` command to halt filtering on messages containing specific characteristics.)

Related concepts

[Policy Script 5: Quarantine External Blacklisted Messages](#) (page 326)

4.3.7 Adding a Disclaimer

Many corporate sites require that all outgoing email have a legal disclaimer attached. PureMessage centralizes this action at the mail gateway.

```
# Add a disclaimer filter.
# Adds "banner.txt" file content to the message body.
# Only detect outgoing mail.
if pmx_relay :memberof "internal-hosts" {
  pmx_add_banner :body :use_html_pre :file "banner.txt";
}
```

Description:

- If the `pmx_relay` test finds the message originated from a relay defined in the “internal-hosts” list:
 - The `pmx_add_banner` action adds the contents of the `banner.txt` file to the body of the message. For HTML messages, this command wraps the file in `<pre>` tags, making the banner look more like plain text. If the message does not contain any text, the banner is added as an attachment to the end of the message.

Where does this filter go?

The “add a disclaimer” filter should be placed before the filter in “Policy Script 1: Scan and Deliver Internal Messages” and directly after the `pmx_test_mark` action. Positioning the “add a disclaimer” filter at this point ensures that:

- The “add a disclaimer” filter always executes on all outgoing mail. (A `stop` command from another filter will not prevent it from executing.)
- Subsequent spam and virus filters will always execute. (The “add a disclaimer” filter does not contain a `stop` command that would otherwise end the policy script before spam and virus filters execute.)

Related concepts

[Policy Script 1: Scan and Deliver Internal Messages](#) (page 324)

5 Regular Expression Primer

The Regular Expressions Primer is a tutorial for those completely new to regular expressions. To familiarize you with regular expressions, this primer starts with the simple building blocks of the syntax and, through examples, builds to construct complex expressions.

PureMessage uses regular expressions as follows:

- The Policy Script: Regular expressions can be used as match operators (“Matches regex” and “Does not match regex”) for many types of policy rule tests. For example, a regular expression can be used to test the contents of a message’s `Envelope to` field. Certain policy rule actions also support the use of regular expressions. For example, the “Deliver immediately for” action supports the use of a regular expression as a match type for exceptions.

For information about using regular expressions while editing the policy script via the PureMessage Manager, see “Policy Rule Tests” and “Policy Rule Actions”. If you are manually editing the policy script, see `pmx-policy` (the documentation for the command-line interface to the PureMessage policy engine).

When regular expressions are used in policy rule tests or actions, they are not prefixed or suffixed with slashes or braces. However, if you are manually editing the policy script on the command-line, you must “escape” backslashes and quotes within regular expressions by preceding them with a backslash. (The PureMessage Manager will automatically escape these characters.) For example, to search for the string “and/or”, enter “and\or” as the regular expression. See “Searching for Special Characters” for more information.

- Lists: Lists can be configured to contain regular expressions. When creating a new list, specify the “Regular Expression” match type. Individual entries in the list are then entered as regular expressions. They are not prefixed with slashes or braces. It is not necessary to escape special characters in regular expressions contained in lists. For more information about configuring lists, see “Lists”.
- Anti-Spam Rules: Anti-spam rules support regular expressions as the test portion of a rule. For example, the content of a message can be tested against a specified regular expression. Many of the default anti-spam rules supplied with PureMessage are based on regular expressions; custom anti-spam rules exclusively use regular expressions. For information about configuring anti-spam rules, see “Spam Detection”.

5.1 About Regular Expressions

Regular expressions are used to describe patterns of characters that match against text strings. They can be used as a tool to search for and replace text, manipulate data, or test for a certain condition in a string of characters. Many everyday tasks can be accomplished with regular expressions, such as checking for the occurrence of a specific word or phrase in the body of an e-mail message, or finding specific file types, such as `.txt` files, in a folder or directory. Regular expressions are often called “regex”, “regexes”, “regexps”, and “RE”. This primer uses the terms “regular expressions”, “regex”, and “regexes” equally.

About Regex Syntax

Regular expressions use syntax elements comprised of alphanumeric characters and symbols. For example, the regex `(2)` searches for the number 2, while the regex `([1-9][0-9]{2}-[0-9]{4})` matches a regular 7-digit phone number.

There are many flavors and types of regular expression syntax. These variations are found in various tools, languages and operating systems. For example, Perl, Python, Tcl, grep, sed, vi, and Unix all use variations on standard regex syntax. This primer focuses on standard regex patterns not tied to a specific language or tool. This standard syntax can be later applied to the specific language, tool or application of your choice.

5.2 Building Simple Patterns

Complete regular expressions are constructed using characters as small building block units. Each building block is in itself simple, but since these units can be combined in an infinite number of ways, knowing how to combine them to achieve a goal takes some practice. This section shows you how to build regexes through examples ranging from the simple to the more complex.

5.2.1 Matching Simple Strings

The simplest and most common type of regex is an alphanumeric string that matches itself, called a “literal text match”. A literal text regex matches anywhere along a string. For example, a literal string matches itself when placed alone, and at the beginning, middle, or end of a larger string. Literal text matches are case sensitive.

Using regexes to search for simple strings.

Example 1: Search for the string “at”.

- **Regex:**

```
at
```

- **Matches:**

```
at  
math  
hat  
ate
```

- **Doesn't Match:**

```
it  
a-t  
At
```

Example 2: Search for the string “email”.

- **Regex:**

```
email
```

- **Matches:**

```
email  
emailing  
many_emails
```

- Doesn't Match:

```
Email
EMAILing
e-mails
```

Example 3: Search for the string "abcdE567".

- Regex:

```
abcdE567
```

- Matches:

```
abcdE567
AabcdE567ing
text_abcdE567
```

- Doesn't Match:

```
SPAMabCdE567
ABCDDe567
```

Note

Regular expressions are case sensitive unless case is deliberately modified.

5.2.2 Searching with Wildcards

In the previous examples, regular expressions are constructed with literal characters that match themselves. There are other characters in regex syntax that match in a more generalized way. These are called "metacharacters". Metacharacters do not match themselves, but rather perform a specific task when used in a regular expression. One such metacharacter is the dot ".", or wildcard. When used in a regular expression, the wildcard can match any single character.

Using the wildcard to match any character.

Example 1: Search for the string "ubject".

- Regex:

```
.ubject:
```

- Matches:

```
Subject:
subject:
Fubject:
```

- Doesn't Match:

```
Subject
subject
```

Example 2: Use three dots "..." to search for any three characters within a string.

- **Regex:**

```
t...s
```

- **Matches:**

```
trees
tEENs
t345s
t-4-s
```

- **Doesn't Match:**

```
Trees
twentys
t1234s
```

Example 3: Use several wildcards to match characters throughout a string.

- **Regex:**

```
.a.a.a
```

- **Matches:**

```
Canada
alabama
banana
3a4a5a
```

- **Doesn't Match:**

```
aaa
```

5.2.3 Searching for Special Characters

In regular expression syntax, most non-alphanumeric characters are treated as special characters. These characters, called “metacharacters”, include asterisks, question marks, dots, slashes, and other non-alphanumeric characters. In order to search for a metacharacter without using its special attribute, precede it with a backslash “\” to change it into a literal character. For example, to build a regex to search for a `.txt` file, precede the dot with a backslash `\.txt` to prevent the dot’s special function, a wildcard search. The backslash, called an “escape character” in regex terminology, turns metacharacters into literal characters.

Precede the following metacharacters with a backslash “\” to search for them as literal characters:

```
^ $ + * ? . | ( ) { } [ ] \
```

Using the backslash “\” to escape special characters in a regular expression.

Example 1: Escape the dollar sign “\$” to find the alphanumeric string “\$100”.

- **Regex:**

```
\$100
```

- **Matches:**

```
$100
$1000
```

- **Doesn't Match:**

```
2100
100
```

Example 2: Use the dot “.” as a literal character to find a file called “email.txt”.

- **Regex:**

```
email\.txt
```

- **Matches:**

```
email.txt
```

- **Doesn't Match:**

```
email
txt
email_txt
```

Example 3: Escape the backslash “\” character to search for a Windows file.

- **Regex:**

```
c:\\readme\\.txt
```

- **Matches:**

```
c:\readme.txt
```

- **Doesn't Match:**

```
c:\\readme.txt
d:\readme.txt
c:/readme.txt
```

5.2.4 Ranges and Repetition

Regex syntax includes metacharacters which specify the number of times a particular character or string must match. This group of metacharacters are called “quantifiers”; they influence the quantity of matches found. Quantifiers act on the element immediately preceding them, which could be a digit, a letter, or another metacharacter (including spaces as metacharacters not previously defined and the dot “.”). This section demonstrates how quantifiers search using ranges and repetition.

Ranges, {min, max}

Ranges are considered “counting qualifiers” in regular expressions. This is because they specify the minimum number of matches to find and the maximum number of matches to allow. Use ranges in regex searches when a bound, or a limit, should be placed on search results. For example, the range {3, 5} matches an item at least 3 times, but not more than 5 times. When this range is combined with the regex, a{3, 5}, the strings “aaa”, “aaaa”, and “aaaaa” are successfully matched. If only a single number is expressed within curly braces {3}, the pattern matches exactly three items. For example, the regex b{3} matches the string “bbb”.

Using ranges to identify search patterns.

Example 1: Match the preceding “0” at least 3 times with a maximum of 5 times.

- **Regex:**

```
60{3,5} years
```

- **Matches:**

```
6000 years
60000 years
600000 years
```

- **Doesn't Match:**

```
60 years
600 years
6003 years
6000000 years
```

Example 2: Using the “.” wildcard to match any character sequence two or three characters long.

- **Regex:**

```
.{2,3}
```

- **Matches:**

```
404
44
com
w3
```

- **Doesn't Match:**

```
4
a
aaaa
```

Example 3: Match the preceding “e” exactly twice.

- **Regex:**

```
be{2}t
```


- Matches:

```
beet
```

- Doesn't Match:

```
bet
beat
eee
```

Example 4: Match the preceding “w” exactly three times.

- Regex:

```
w{3}\.mydomain\.com
```

- Matches:

```
www.mydomain.com
```

- Doesn't Match:

```
web.mydomain.com
w3.mydomain.com
```

Repetition, ? * +

Unlike range quantifiers, the repetition quantifiers (question mark “?”, asterisk “*”, and plus “+”) have few limits when performing regex searches, they are greedy. This is significant because these quantifiers settle for the minimum number of required matches, but always attempt to match as many times as possible, up to the maximum allowed. For example, the question mark “?” matches any preceding character 0 or 1 times, the asterisk “*” matches the preceding character 0 or more times, and the plus “+” matches the preceding character 1 or more times. Use repetition quantifiers in regex searches when large numbers of results are desired.

Using repetition to search for repeated characters with few limits.

Example 1: Use “?” to match the “u” character 0 or 1 times.

- Regex:

```
colou?r
```

- Matches:

```
colour
color
```

- Doesn't Match:

```
colouur
Colour
```

Example 2: Use “*” to match the preceding item 0 or more times; use “.” to match any character.

- **Regex:**

```
www\.my.*\.com
```

- **Matches:**

```
www.mysite.com
www.mypage.com
www.my.com
```

- **Doesn't Match:**

```
www.oursite.com
mypage.com
```

Example 3: Use “+” to match the preceding “5” at least once.

- **Regex:**

```
bob5+@foo\.com
```

- **Matches:**

```
bob5@foo.com
bob5555@foo.com
```

- **Doesn't Match:**

```
bob@foo.com
bob65555@foo.com
```

Quantifier Summary

The following table defines the various regex quantifiers. Note that each quantifier is unique and will perform a varying minimum and maximum number of matches in order to search successfully.

Quantifier	Description
<code>{ num }</code>	Matches the preceding element <i>num</i> times.
<code>{ min , max }</code>	Matches the preceding element at least <i>min</i> times, but not more than <i>max</i> times.
<code>?</code>	Matches any preceding element 0 or 1 times.
<code>*</code>	Matches the preceding element 0 or more times.
<code>+</code>	Matches the preceding element 1 or more times.

5.2.5 Using Conditional Expressions

Conditional expressions help qualify and restrict regex searches, increasing the probability of a desirable match. The vertical bar “|” symbol, meaning “OR”, places a condition on the regex to search for either one character in a string or another. Because the regex has a list of alternate choices to evaluate, this regex technique is called “alternation”. To search for either one character or another, insert a vertical bar “|” between the desired characters.

Example 1: Use “|” to alternate a search for various spellings of a string.

- **Regex:**

```
gray|grey
```

- **Matches:**

```
gray
grey
```

- **Doesn't Match:**

```
GREY
Gray
```

Example 2: Use “|” to alternate a search for either email or Email or EMAIL or e-mail.

- **Regex:**

```
email|Email|EMAIL|e-mail
```

- **Matches:**

```
email
Email
EMAIL
e-mail
```

- **Doesn't Match:**

```
EmAiL
E-Mail
```

5.2.6 Grouping Similar Items in Parentheses

Use parentheses to enclose a group of related search elements. Parentheses limit scope on alternation and create substrings to enhance searches with metacharacters. For example, use parentheses to group the expression `(abc)`, then apply the range quantifier `{3}` to find instances of the string “abcabcabc”.

Using parentheses to group regular expressions.

Example 1: Use parentheses and a range quantifier to find instances of the string “abcabcabc”.

- **Regex:**

```
(abc){3}
```

- **Matches:**

```
abcabcabc
abcabcabcabc
```

Note

In the second match, the match will actually be to the first nine characters only.

- Doesn't Match:

```
abc
abcabc
```

Example 2: Use parentheses to limit the scope of alternative matches on the words gray and grey.

- Regex:

```
gr(a|e)y
```

- Matches:

```
gray
grey
```

- Doesn't Match:

```
gry
graey
```

Example 3: Use parentheses and “|” to locate past correspondence in a mail-filtering program. This regex finds a “To:” or a “From:” line followed by a space and then either the word “Smith” or the word “Chan”.

- Regex:

```
(To:|From:)(Smith|Chan)
```

- Matches:

```
To:Smith
To:Chan
From:Smith
To:Smith, Chan
To:Smithe
From:Channel4News
```

- Doesn't Match:

```
To:smith
To:All
To:Schmidt
```

5.2.7 Matching Sequences

You can build a regular expression to match a sequence of characters. These sequences, called “character classes”, simply place a set of characters side-by-side within square brackets “[]”. An item in a character class can be either an ordinary character, representing itself, or a metacharacter, performing a special function. This primer covers how to build simple character classes, prevent matches with character classes, and construct compound character classes with metacharacters.

Building Simple Character Classes

The most basic type of character class is a set of characters placed side-by-side within square brackets "[]". For example, the regular expression `[bcr]at`, matches the words "bat", "cat", or "rat" because it uses a character class (that includes "b", "c", or "r") as its first character. Character classes only match singular characters unless a quantifier is placed after the closing bracket. For examples using quantifiers with character classes, see [Compound Character Classes](#). The following table shows how to use simple character classes in regex searches.

Note

When placed inside a character class, the hyphen "-" metacharacter denotes a continuous sequence of letters or numbers in a range. For example, `[a-d]` is a range of letters denoting the continuous sequence of a,b,c and d. When a hyphen is otherwise used in a regex, it matches a literal hyphen.

Using simple character classes to perform regex searches.

Example 1: Use a character class to match all cases of the letter "s".

- **Regex:**

```
Java[Ss]cript
```

- **Matches:**

```
JavaScript
Javascript
```

- **Doesn't Match:**

```
javascript
javaScript
```

Example 2: Use a character class to limit the scope of alternative matches on the words gray and grey.

- **Regex:**

```
gr[ae]y
```

- **Matches:**

```
gray
grey
```

- **Doesn't Match:**

```
gry
graey
```

Example 3: Use a character class to match any one digit in the list.

- **Regex:**

```
[0123456789]
```

- Matches:

```
5
0
9
```

- Doesn't Match:

```
x
?
F
```

Example 4: To simplify the previous example, use a hyphen “-” within a character class to denote a range for matching any one digit in the list.

- Regex:

```
[0-9]
```

- Matches:

```
5
0
9
```

- Doesn't Match:

```
234
42
```

Example 5: Use a hyphen “-” within a character class to denote an alphabetic range for matching various words ending in “mail”.

- Regex:

```
[A-Z]mail
```

- Matches:

```
Email
Xmail
Zmail
```

- Doesn't Match:

```
email
mail
```

Example 6: Match any three or more digits listed in the character class.

- Regex:

```
[0-9]{3,}
```

- Matches:

```
012
1234
555
98754378623
```

- Doesn't Match:

```
10
7
```

Preventing Matches with Character Classes

Previous examples used character classes to specify exact sequences to match. Character classes can also be used to prevent, or negate, matches with undesirable strings. To prevent a match, use a leading caret “^” (meaning NOT), within square brackets, [^ . . .]. For example, the regex [^a] matches any single character except the letter “a”.

Note

The caret symbol must be the first character within the square brackets to negate a character class.

Using character classes to prevent a sequence from matching.

Example 1: Prevent a match on any numeric string. Use the “*” to match an item 0 or more times.

- Regex:

```
[^0-9]*
```

- Matches:

```
abc
c
Mail
u-see
a4a
```

- Doesn't Match:

```
1
42
100
23000000
```

Example 2: Search for a text file beginning with any character not a lower-case letter.

- Regex:

```
[^a-z]\.txt
```

- Matches:

```
A.txt
4.txt
Z.txt
```

- Doesn't Match:

```
r.txt  
a.txt  
Aa.txt
```

Example 3: Prevent a match on the numbers "10" and "12".

- Regex:

```
1[ ^02]
```

- Matches:

```
13  
11  
19  
17  
1a
```

- Doesn't Match:

```
10  
12  
42  
a1
```

Compound Character Classes

Character classes are a versatile tool when combined with various pieces of the regex syntax. Compound character classes can help clarify and define sophisticated searches, test for certain conditions in a program, and filter wanted e-mail from spam. This section uses compound character classes to build meaningful expressions with the regex syntax.

Using compound character classes with the regex syntax.

Example 1: Find a partial e-mail address. Use a character class to denote a match for any number between 0 and 9. Use a range to restrict the number of times a digit matches.

- Regex:

```
smith[0-9]{2}@
```

- Matches:

```
smith44@  
smith42@
```

- Doesn't Match:

```
Smith34  
smith6  
Smith0a
```

Example 2: Search an HTML file to find each instance of a header tag. Allow matches on whitespace after the tag but before the ">".

- **Regex:**

```
(<[Hh] [1-6] *>)
```

- **Matches:**

```
<H1>
<h6>
<H3  >
<h2    >
```

- **Doesn't Match:**

```
<H1
<    h2>
<a1>
```

Example 3: Match a regular 7-digit phone number. Prevent the digit "0" from leading the string.

- **Regex:**

```
([1-9] [0-9]{2} - [0-9]{4})
```

- **Matches:**

```
555-5555
123-4567
```

- **Doesn't Match:**

```
555.5555
1234-567
023-1234
```

Example 4: Match a valid web-based protocol. Escape the two front slashes.

- **Regex:**

```
[a-z]+:\\\\
```

- **Matches:**

```
http://
ftp://
tcl://
https://
```

- **Doesn't Match:**

```
http
http:
1a3://
```

Example 5: Match a valid e-mail address.

- **Regex:**

```
[a-z0-9_-]+(\\.[a-z0-9_-]+)*@[a-z0-9_-]+(\\.[a-z0-9_-]+)+
```

- Matches:

```
j_smith@foo.com
j.smith@bc.canada.ca
smith99@foo.co.uk
1234@mydomain.net
```

- Doesn't Match:

```
@foo.com
.smith@foo.net
smith.@foo.org
www.myemail.com
```

Note

This regular expression will actually also match the `smith@foo.net` part of the `.smith@foo.net` example.

Character Class Summary

The following table defines various character class sequences. Use these alphanumeric patterns to simplify your regex searches.

Character Class	Description
[0-9]	Matches any digit from 0 to 9.
[a-zA-z]	Matches any alphabetic character.
[a-zA-z0-9]	Matches any alphanumeric character.
[^0-9]	Matches any non-digit.
[^a-zA-z]	Matches any non-alphabetic character.

5.2.8 Matching Locations within a String

At times, the pattern to be matched appears at either the very beginning or end of a string. In these cases, use a caret “^” to match a desired pattern at the beginning of a string, and a dollar sign “\$” for the end of the string. For example, the regular expression `email` matches anywhere along the following strings: “email”, “emailing”, “bogus_emails”, and “smithsemailaddress”. However, the regex `^email` only matches the strings “email” and “emailing”. The caret “^” in this example is used to effectively anchor the match to the start of the string. For this reason, both the caret “^” and dollar sign “\$” are referred to as anchors in the regex syntax.

Note

The caret “^” has many meanings in regular expressions. Its function is determined by its context. The caret can be used as an anchor to match patterns at the beginning of a string, for example: `(^File)`. The caret can also be used as a logical “NOT” to negate content in a character class, for example: `[^...]`.

Using anchors to match at the beginning or end of a string.

Example 1: Use "\$" to match the ".com" pattern at the end of a string.

- **Regex:**

```
.*\.com$
```

- **Matches:**

```
mydomain.com
a.b.c.com
```

- **Doesn't Match:**

```
mydomain.org
mydomain.com.org
```

Example 2: Use "^" to match "inter" at the beginning of a string, "\$" to match "ion" at the end of a string, and "." to match any number of characters within the string.

- **Regex:**

```
^inter.*ion$
```

- **Matches:**

```
internationalization
internalization
```

- **Doesn't Match:**

```
reinternationalization
```

Example 3: Use "^" inside parentheses to match "To" and "From" at the beginning of the string.

- **Regex:**

```
(^To:|^From:) (Smith|Chan)
```

- **Matches:**

```
From:Chan
To:Smith
From:Smith
To:Chan
```

- **Doesn't Match:**

```
From: Chan
from:Smith
To Chan
```

Example 4: Performing the same search as #3, place the caret "^" outside the parentheses this time for similar results.

- Regex:

```
^(From|Subject|Date):(Smith|Chan|Today)
```

- Matches:

```
From:Smith  
Subject:Chan  
Date:Today
```

- Doesn't Match:

```
X-Subject:  
date:Today
```

5.3 More Regex Resources

Note

Regular expressions are very powerful tools and they don't always work quite the way that you would expect. As a result, they can also be dangerous. An untested regular expression can fail to match what you want it to match, and it can match what you don't expect it to match. To ensure safe use of your regular expressions, always test them, and never test them on live data. The links provided below represent some of the best tutorials and resources for learning and working with regular expressions that are available on the Internet; however, you must always test any of their examples or any of the regular expressions you develop based on their tutorials on sample data.

Beginner:

- [Perl Regular Expressions Reference](#), ActiveState Programmer Network (ASPN)
- [Perl Regular Expressions Tutorial](#), ActiveState Programmer Network (ASPN)
- [Five Habits for Successful Regular Expressions](#), The O'Reilly ONLamp Resource Center
- [Beginner's Introduction to Perl - Part 3](#), The O'Reilly Perl Resource Center

Intermediate:

- [Rx Cookbook](#), ActiveState Programmer Network (ASPN)
- [Regexp Power](#), The O'Reilly Perl Resource Center

Advanced:

- [Power Regexp](#), Part II, The O'Reilly Perl Resource Center

Examples:

- <http://regexlib.com/> provides numerous regular expression examples, including regular expressions for matching emails and URIs.

6 Copyrights and Trademarks

Copyright © 2000-2011 Sophos Limited. All rights reserved. Sophos and PureMessage are trademarks of Sophos Limited. All other trademarks are trademarks or registered trademarks of their respective owners.

This Sophos software is licensed in accordance with the terms of the Sophos End User License Agreement. A copy of this license agreement can be found at <http://www.sophos.com/legal>

This software includes or may include:

- * Perl and Perl modules originally written by others. Perl and all CPAN modules are used in this Sophos software in accordance with the terms of the Perl Artistic License. A copy of this license agreement is available at <http://www.perl.com/pub/a/language/misc/Artistic.html>

The source code for Perl is available at <http://www.cpan.org>

- * Software developed by the OpenSSL Project for use in the OpenSSL Toolkit is available at <http://www.openssl.org>

- * Cryptographic software written by Eric A. Young and software written by Tim J. Hudson.

- * Software originally written by Philip Hazel. Copyright © The University of Cambridge, England. The source code for this software is available at: <ftp://ftp.csx.cam.ac.uk/pub/software/programming/pcrc/>

- * Software developed by the Apache Software Foundation (<http://www.apache.org/>) and the Apache SpamAssassin Project. A copy of the license agreement for this software can be found at <http://www.apache.org/licenses/LICENSE-2.0.txt>

- * Software developed by The OpenLDAP Foundation. Copyright © 1999-2003 The OpenLDAP Foundation, Redwood City, California, USA. A copy of the license agreement for this software is reproduced below.

- * Sendmail software. Copyright © 1988, 1993 The Regents of the University of California. A copy of the license agreement for this software is reproduced below.

- * Postfix software. The source code for this software is available at: <http://www.postfix.org/download.html>. Changes made by Sophos are available via mail order by submitting a request to Sophos; via email to support@sophos.com or via the web at <http://www.sophos.com/support/queries/enterprise.html>.

- * Software originally written by Thomas G. Lane. This software is based in part on the work of the Independent JPEG Group.

- * Software originally written by David Turner, Robert Wilhelm, and Werner Lemberg. Portions of this software are copyright © 2006 The FreeType Project (www.freetype.org).

- * Software originally written by the zlib team, Jean-loup Gailly & Mark Adler.

- * Software originally written by Thomas Boutell, Portions copyright 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004 by Cold Spring Harbor Laboratory. Funded under Grant P41-RR02188 by the National Institutes of Health, Portions copyright 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004 by Boutell.Com, Inc., Portions relating to GD2 format copyright 1999, 2000, 2001, 2002, 2003, 2004 Philip Warner, Portions relating to PNG copyright 1999, 2000, 2001, 2002, 2003, 2004 Greg Roelofs, Portions relating to gdtf.c copyright 1999, 2000, 2001, 2002, 2003, 2004 John Ellson (ellson@graphviz.org), Portions relating to gdtf.c copyright 2001, 2002, 2003, 2004 John Ellson (ellson@graphviz.org), Portions relating to JPEG and to color quantization copyright 2000, 2001, 2002, 2003, 2004, Doug Becker and copyright © 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004 Thomas G. Lane. This software is based in part on the work of the Independent JPEG Group. See the file README-JPEG.TXT for more information, Portions relating to GIF compression

copyright 1989 by Jef Poskanzer and David Rowley, with modifications for thread safety by Thomas Boutell, Portions relating to GIF decompression copyright 1990, 1991, 1993 by David Koblas, with modifications for thread safety by Thomas Boutell, Portions relating to WBMP copyright 2000, 2001, 2002, 2003, 2004 Maurice Szmurlo and Johan Van den Brande, Portions relating to GIF animations copyright 2004 Jaakko Hyvätti (jaakko.hyvatti@iki.fi)

* Some software programs that are licensed (or sublicensed) to the user under the GNU General Public License (GPL) or other similar Free Software licenses, which, among other rights, permit the user to copy, modify and redistribute certain programs, or portions thereof, and have access to the source code. The GPL requires for any software licensed under the terms of the GPL, which is distributed in an executable binary format, that the source code for such software also be made available to the users of the binary form. For any such software covered under the GPL, the source code is available via mail order by submitting a request to Sophos; via email to support@sophos.com or via the web at <http://www.sophos.com/support/queries/enterprise.html>. A copy of the license agreement for any such included software can be found at <http://www.gnu.org/copyleft/gpl.html>.

* Socket 6

Copyright (C) 2000-2008 Hajimu UMEMOTO ume@mahoroba.org. All rights reserved.

Socket6.pm and Socket6.xs are based on perl5.005_55-v6-19990721 written by KAME Project.

gai.h, getaddrinfo.c and getnameinfo.c are based on ssh-1.2.27-IPv6-1.5 written by KIKUCHI Takahiro kick@kyoto.wide.ad.jp.

Copyright (C) 1995, 1996, 1997, 1998, and 1999 WIDE Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the project nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE PROJECT AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE PROJECT OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* LibDB

* Copyright (c) 1990, 1993

* The Regents of the University of California. All rights reserved.

*

* This code is derived from software contributed to Berkeley by

* Chris Torek.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.

* 3. All advertising materials mentioning features or use of this software
 * must display the following acknowledgement:
 * This product includes software developed by the University of
 * California, Berkeley and its contributors.

* 4. Neither the name of the University nor the names of its contributors
 * may be used to endorse or promote products derived from this software
 * without specific prior written permission.

* THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.

* OpenSSL

* =====

* Copyright (c) 1998-2011 The OpenSSL Project. All rights reserved.

* Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:

* 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.

* 3. All advertising materials mentioning features or use of this
 * software must display the following acknowledgment:
 * "This product includes software developed by the OpenSSL Project
 * for use in the OpenSSL Toolkit. <http://www.openssl.org/>

* 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
 * endorse or promote products derived from this software without
 * prior written permission. For written permission, please contact OpenSSL at
 * openssl-core@openssl.org.

* 5. Products derived from this software may not be called "OpenSSL"
 * nor may "OpenSSL" appear in their names without prior written
 * permission of the OpenSSL Project.

```
* 6. Redistributions of any form whatsoever must retain the following
* acknowledgment:
* "This product includes software developed by the OpenSSL Project
* for use in the OpenSSL Toolkit http://www.openssl.org/"
*
* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY
* EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
* PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
* OF THE POSSIBILITY OF SUCH DAMAGE.
* =====
*
* This product includes cryptographic software written by Eric Young
* (eay@cryptsoft.com). This product includes software written by Tim
* Hudson (tjh@cryptsoft.com).
*
*
```

Original SSLeay License

```
-----
* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
* All rights reserved.
*
* This package is an SSL implementation written
* by Eric Young (eay@cryptsoft.com).
* The implementation was written so as to conform with Netscapes SSL.
*
* This library is free for commercial and non-commercial use as long as
* the following conditions are aheared to. The following conditions
* apply to all code found in this distribution, be it the RC4, RSA,
* lhash, DES, etc., code; not just the SSL code. The SSL documentation
* included with this distribution is covered by the same copyright terms
* except that the holder is Tim Hudson (tjh@cryptsoft.com).
*
* Copyright remains Eric Young's, and as such any Copyright notices in
* the code are not to be removed.
* If this package is used in a product, Eric Young should be given attribution
* as the author of the parts of the library used.
* This can be in the form of a textual message at program startup or
* in documentation (online or textual) provided with the package.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
```


- * documentation and/or other materials provided with the distribution.
- * 3. All advertising materials mentioning features or use of this software
- * must display the following acknowledgement:
- * "This product includes cryptographic software written by
- * Eric Young (eay@cryptsoft.com)"
- * The word 'cryptographic' can be left out if the routines from the library
- * being used are not cryptographic related :-).
- * 4. If you include any Windows specific code (or a derivative thereof) from
- * the apps directory (application code) you must include an acknowledgement:
- * "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
- *
- * THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND
- * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
- * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
- * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
- * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
- * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
- * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
- * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
- * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
- * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
- * SUCH DAMAGE.
- *
- * The licence and distribution terms for any publically available version or
- * derivative of this code cannot be changed. i.e. this code cannot simply be
- * copied and put under another distribution licence
- * [including the GNU Public Licence.]
- *

* libxml2

Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar licence but with different Copyright notices) all the files are:

Copyright (C) 1998-2003 Daniel Veillard. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.

* PostgreSQL

PostgreSQL is released under the [PostgreSQL License](#), a liberal Open Source license, similar to the BSD or MIT licenses.

PostgreSQL Database Management System (formerly known as Postgres, then as Postgres95)

Portions Copyright (c) 1996-2010, The PostgreSQL Global Development Group

Portions Copyright (c) 1994, The Regents of the University of California

Permission to use, copy, modify, and distribute this software and its documentation for any purpose, without fee, and without a written agreement is hereby granted, provided that the above copyright notice and this paragraph and the following two paragraphs appear in all copies.

IN NO EVENT SHALL THE UNIVERSITY OF CALIFORNIA BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF THE UNIVERSITY OF CALIFORNIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE UNIVERSITY OF CALIFORNIA SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE UNIVERSITY OF CALIFORNIA HAS NO OBLIGATIONS TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

* Some software programs that are licensed (or sublicensed) to the user under the GNU General Public License (GPL) or other similar Free Software licenses, which, among other rights, permit the user to copy, modify and redistribute certain programs, or portions thereof, and have access to the source code. The GPL requires for any software licensed under the terms of the GPL, which is distributed in an executable binary format, that the source code for such software also be made available to the users of the binary form. For any such software covered under the GPL, the source code is available via mail order by submitting a request to Sophos; via email to support@sophos.com or via the web at <http://www.sophos.com/support/queries/enterprise.html>. A copy of the license agreement for any such included software can be found at <http://www.gnu.org/copyleft/gpl.html>.

6.1 OpenLDAP License

The OpenLDAP Public License Version 2.8, 17 August 2003 Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met:

1. Redistributions in source form must retain copyright statements and notices, 2. Redistributions in binary form must reproduce applicable copyright statements and notices, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution, and 3. Redistributions must contain a verbatim copy of this document. The OpenLDAP Foundation may revise this license from time to time. Each revision is distinguished by a version number. You may use this Software under terms of this license revision or under the terms of any subsequent revision of the license.

THIS SOFTWARE IS PROVIDED BY THE OPENLDAP FOUNDATION AND ITS CONTRIBUTORS "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OPENLDAP FOUNDATION, ITS CONTRIBUTORS, OR THE AUTHOR(S) OR OWNER(S) OF THE SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The names of the authors and copyright holders must not be used in advertising or otherwise to promote the sale, use or other dealing in this Software without specific, written prior permission. Title to copyright in this Software shall at all times remain with copyright holders. OpenLDAP is a registered trademark of the OpenLDAP Foundation. Copyright 1999-2003 The OpenLDAP Foundation, Redwood City, California, USA. All Rights Reserved. Permission to copy and distribute verbatim copies of this document is granted.

6.2 SEE License

The SEE library source is released under what is commonly called a "BSD-style" licence:

```
/*
 * Copyright (c) 2003, 2004, 2005, 2006, 2007
 *   David Leonard. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *   notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *   notice, this list of conditions and the following disclaimer in the
 *   documentation and/or other materials provided with the distribution.
 * 3. Neither the name of David Leonard nor the names of its contributors
 *   may be used to endorse or promote products derived from this software
 *   without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
 * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
 * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
 * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
 * COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,
 * BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
 * CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
 * ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
 * POSSIBILITY OF SUCH DAMAGE.
 */
```

The separate 'dtoa.c' file is separately licenced, thus:

```
/*
 * *****
 *
 * The author of this software is David M. Gay.
 *
 * Copyright (c) 1991, 2000 by Lucent Technologies.
 *
 * Permission to use, copy, modify, and distribute this software for any
 * purpose without fee is hereby granted, provided that this entire notice
 * is included in all copies of any software which is or includes a copy
 * or modification of this software and in all copies of the supporting
```

```
* documentation for such software.  
*  
* THIS SOFTWARE IS BEING PROVIDED "AS IS", WITHOUT ANY EXPRESS OR IMPLIED  
* WARRANTY. IN PARTICULAR, NEITHER THE AUTHOR NOR LUCENT MAKES ANY  
* REPRESENTATION OR WARRANTY OF ANY KIND CONCERNING THE MERCHANTABILITY  
* OF THIS SOFTWARE OR ITS FITNESS FOR ANY PARTICULAR PURPOSE.  
*  
*****/
```

6.3 Sendmail License

The following license terms and conditions apply, unless a different license is obtained from Sendmail, Inc., 6425 Christie Ave, Fourth Floor, Emeryville, CA 94608, USA, or by electronic mail at license@sendmail.com. License Terms:

Use, Modification and Redistribution (including distribution of any modified or derived work) in source and binary forms is permitted only if each of the following conditions is met:

1. Redistributions qualify as "freeware" or "Open Source Software" under one of the following terms:

- (a) Redistributions are made at no charge beyond the reasonable cost of materials and delivery.
- (b) Redistributions are accompanied by a copy of the Source Code or by an irrevocable offer to provide a copy of the Source Code for up to three years at the cost of materials and delivery. Such redistributions must allow further use, modification, and redistribution of the Source Code under substantially the same terms as this license. For the purposes of redistribution "Source Code" means the complete compilable and linkable source code of sendmail including all modifications.

2. Redistributions of source code must retain the copyright notices as they appear in each source code file, these license terms, and the disclaimer/limitation of liability set forth as paragraph 6 below.

3. Redistributions in binary form must reproduce the Copyright Notice, these license terms, and the disclaimer/limitation of liability set forth as paragraph 6 below, in the documentation and/or other materials provided with the distribution. For the purposes of binary distribution the "Copyright Notice" refers to the following language: "Copyright © 1998-2004 Sendmail, Inc. All rights reserved."

4. Neither the name of Sendmail, Inc. nor the University of California nor the names of their contributors may be used to endorse or promote products derived from this software without specific prior written permission. The name "sendmail" is a trademark of Sendmail, Inc.

5. All redistributions must comply with the conditions imposed by the University of California on certain embedded code, whose copyright notice and conditions for redistribution are as follows:

- (a) Copyright © 1988, 1993 The Regents of the University of California. All rights reserved.
- (b) Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
 - (i) Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
 - (ii) Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
 - (iii) Neither the name of the University nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

6. Disclaimer/Limitation of Liability: THIS SOFTWARE IS PROVIDED BY SENDMAIL, INC. AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL SENDMAIL, INC., THE

REGENTS OF THE UNIVERSITY OF CALIFORNIA OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Note

You can find the source code at <ftp://ftp.sendmail.org/pub/sendmail>

7 Installing Sender History Database and Configuring Delay Queue

To install Sender History Database role and configure Delay Queue:

1. Installation of Sender History Database Role.

a. Installation

1. Use the PMX installer. It will ask for permission to update itself. Answer **Yes**.
2. Select **Custom Installation**.
3. Select **Sender History Database**.
4. Specify the email address for notification messages.
5. Specify the mail server to send notifications to.
6. Specify the directory in which the sender history database file should be stored.
7. Specify the internal host IP address/subnet, if any.
8. Proceed with the installation.

Note

If you have **Full installations** and want to upgrade to 6.4.0:

- Upgrade to 6.4.0 (new packages for existings roles will be upgraded).
- Run the installer, go to **Add Additional Components**, select **Sender History Database Role** and install.

b. Configuration

Sender history database is maintained using Redis (<https://redis.io/>). You need to configure: `<pmx_directory>/etc/redis.conf`. Explanation of parameters is given below. Check the values of: bind, port, dir and dbfilename.

Basic configuration: Network

Specify the network interface on which Redis should listen for connections. Default is to listen on all the available network interfaces. This configuration is not secure if the system is exposed to Internet. If you choose to listen on all interfaces (we DO NOT recommend this configuration, if the system is exposed to internet), set protected-mode to no. Default port is 6379 .

```
bind 192.168.56.29
port 6379
```

Working directory location, database will be saved here.

```
dir /opt/pmx6/var/redis
```

The name of the database file (this file will be created in the directory specified above).

```
dbfilename edge.domain-c.vbox_shdb.rdb
```

Advance configuration (related to performance, fine tune this configuration only if required)

Backup frequency (when to save the database on disk).

save at every 7 minutes, irrespective of keys changed.

```
save 420 1
```

save at every 4 minutes if (at least) 500 keys changed.

```
save 240 500
```

save at every minute if (at least) 1300 keys changed.

```
save 60 1000
```

Maximum number of simultaneous client connections.

Verify if your Linux system (particularly if you are using CentOS) is configured to allow the number of open files `grep 'nofile' /etc/security/limits.conf; cat /proc/`pidof redis-server`/limits|grep "Max open files."`

```
maxclients 15000
```

Do not use AOF method of saving records.

```
appendonly no
```

Note

Verify if the redis service can be reached through the edge server. To ensure this, you may need to adjust firewalling and routing.

Note

Resources system must be initialized (if not, already) on the CSM role for delay queue to work properly. For more information refer:

```
perldoc pmx-resources-init
```

2. Optional: Use CSM to share three configuration files: `redis_location.conf`, `delay.conf` and `policy.siv`

These three files are necessary for delay queue. You can either share them with all edge servers using CSM or you can manage them individually on each edge server.

1. `redis_location.conf`

Each edge server needs to know how the redis server can be contacted in order to set and get sender history database entries. The file `redis_location.conf` should be present on edge server at location: `<pmx_directory>/etc/redis_location.conf`. Syntax of the file is:

```
server_ip = 10.200.193.229
port = 6379
```

Note

Communication between milter and historian DB role will be in plain text. You need to restart the milter service if there is change in this file.

2. `delay.conf`

This file contains parameters that affect the decisions of delaying messages. You need to restart the milter service if there is any change in this file.

Note

Communication between milter and historian DB role will be in plain text. You need to restart the milter service if there is change in this file.

This file should be present on each edge server at the location: `<pmx_directory>/etc/delay.conf`

```
delay_status = Collect
delay_min_minutes = 10
delay_max_minutes = 60
delay_max_size = 1024
send_delayed_mail_to = 127.0.0.1:10025
```

delay_status

Possible values: **Collect**, **On**, and **Off**

Default: Collect

Example: `delay_status = Collect`

Description:

Collect: Do not delay mails based on sender history, instead build the sender history database.

On: Enable snow shoe spam protection

Off: Disable snow shoe spam protection.

delay_min_minutes

Description: minimum time (in minutes) a message can be delayed.

Default: 2

Example: `delay_min_minutes = 10`

Possible values: The value must not be less than 2 and it must not be greater than or equal to `delay_max_minutes`.

delay_max_minutes

Description: maximum time (in minutes) a message can be delayed.

Default: 60

Example: `delay_max_minutes = 60`

Possible values: The value must not be greater than 60 and it must not be less than or equal to `delay_min_minutes`.

```
delay_max_size
```

Description: Delay queue disk size limit (MB).

Default: 1024

Example: `delay_max_size = 1024`

Possible values: The value must not be less than 256 and it must not be greater than 1024.

```
send_delayed_mail_to
```

It describes the destination of delayed message when it is time to re-inject the message into the militer.

For PMX supplied Postfix MTA, refer to the port parameter in the militer section of 'Policy' in the file `pmx.conf`.

```
Example: send_delayed_mail_to = 127.0.0.1:10025
```

For PMX supplied Sendmail MTA, use the IP address and port that Sendmail is receiving mails on.

```
Example: send_delayed_mail_to = 192.168.1.1:25
```

The key here is to use the IP and port so that the mail is reinjected into the same edge server that has processed (and delayed) it before.

Note

Make sure that relay is accepting the mails (mails that are getting re-injected). If not, mails (that will fill the disk upto the value of `delay_max_size`) will stack in the delay queue directory until the size of the directory goes beyond the `delay_max_size` configuration parameter.

3. `policy.siv`

A new test `pmx_delayed_mail` and a new action `pmx_suspect_delay` have been added to be used in `policy.siv`.

The test `pmx_delayed_mail` returns true if the message being processed was delayed earlier and is getting reinjected into the militer. Use this test to prevent the actions and test, which are to be applied only to the message when it first arrived, from being applied to the delayed message. For more details, click [policy.siv](#)

Note

It is recommended to use command `pmx-policy` to edit the `policy.siv` file to search for syntax error. If you are using any other method to edit the policy (example: a text editor), use command `pmx-policy` to check the correctness of the policy.

The action `pmx_suspect_delay` delays the message for a period specified by the antispam engine. The message will be delayed only if the `delay_status` is 'On' in `<pmx_directory>/`

etc/delay.conf file. This action should be used in conjunction with the test pmx_spam_hit like following:

```
if pmx_spam_hit :comparator "i;ascii-casemap" :matches ["DQ_SUSP?"]  
{  
    pmx_suspect_delay;  
}
```

Note

Do not use **stop** action. It is implicit, if the message is delayed, message processing will stop internally.

A sample policy script is given that will help learning how this new action and test should be used.

3. Upgrading the edge servers

Upgrade the edge servers.

7.1 policy.siv

Displays the new actions added in policy.siv.

```
require "PureMessage";

# Mark the subject (for both incoming and outgoing messages)
pmx_mark "s" "%%SUBJECT:h_utf8%%";
# attr NAME=Mail from internal hosts
if pmx_relay :memberof "internal-hosts" {
  if not pmx_delayed_mail {
    # The 'pmx-mlog-watch' depends on this to know which messages
    # are outgoing and which are not.
    pmx_markl "i";
    # attr NAME=Check for mail containing viruses
    if pmx_virus {
      # attr LICENSE=PureMessage::Policy::Virus
      # attr NAME=Allow unscannable messages to pass through
      if pmx_virus_cantscan {
        pmx_replace_header :index 0 "X-PMX-Virus" "Unscannable";
        pmx_replace_header :index 0 "Subject" "[POTENTIAL VIRUS] %
%SUBJECT%%";
        pmx_mark "pmx_reason" "Unscannable";
      }
      # attr NAME=Reject mail containing viruses
      else {
        pmx_mark "pmx_reason" "Virus";
        reject "Virus(es) (%%VIRUS_IDS%%) were detected in the
message.";
        stop;
      }
    }
  }
}
# attr NAME=Mail from external hosts
else {
  if not pmx_delayed_mail {
    pmx_add_header "X-PMX-Version" "%%PMX_VERSION%%";
    pmx_mark "Size" "%%MESSAGE_SIZE%%";
    # attr NAME=Quarantine blocked IP addresses (Sophos Blocklist)
    if pmx_blocklist {
      pmx_mark "pmx_reason" "Block List";
      pmx_quarantine "Blocked";
      stop;
    }
    # attr NAME=Check for mail containing viruses
    if pmx_virus {
      # attr LICENSE=PureMessage::Policy::Virus
      # attr NAME=Allow unscannable messages to pass through
      if pmx_virus_cantscan {
        pmx_replace_header :index 0 "X-PMX-Virus" "Unscannable";
        pmx_replace_header :index 0 "Subject" "[POTENTIAL VIRUS] %
%SUBJECT%%";
        pmx_mark "pmx_reason" "Unscannable";
      }
      # attr NAME=Quarantine mail containing viruses
      else {
        pmx_mark "pmx_reason" "Virus";
        pmx_quarantine "Virus";

```

8 Glossary

8.1 Active Directory

Microsoft implementation of LDAP (Lightweight Directory Access Protocol) for Windows. Active Directory provides LDAP-like directory services for managing identities and permissions of users throughout a network. Active Directory is a hierarchical, object-oriented database in which each object represents a single entity, for example, a user or group.

8.2 blackhole list

A list of hosts known to be sources of spam.

8.3 blacklist

A list used to block mail from specific hosts. In Sophos email and URL filtering products, this list is also referred to as a blocked hosts/senders list. This type of list is also known as a “block list”.

8.4 blocker

A utility that rejects messages originating from IP addresses blacklisted by Sophos Labs and from senders that are otherwise deemed to be spammers, either prior to processing by the MTA or in the policy.

8.5 CSM

The Centralized Server Manager is one of the server roles that is installed in a PureMessage deployment. The CSM server must be installed on the same server as the Database Server role. In a multi-server installation, this server must be set up first.

8.6 Dashboard

The default page in the administrative UI. Shows key statistics and status.

8.7 day zero

The period during which a threat is so new that no threat detection signatures are available to protect against it. Fast moving threats such as internet worms can cause huge amounts of damage at day zero. Sophos uses its Genotype™ technology to block families of spam and viruses, offering a form of protection against previously unseen threats even before specific detection is available.

8.8 denial of service (DOS) attack

An attack on a host or network that causes a loss of service to its users. This is usually done by consuming the bandwidth of the target system or overloading its computational resources with multiple, distributed connections.

8.9 dictionary attack

A technique of trying to guess a user's password by running through a list of likely possibilities, often a list of words from a dictionary. It contrasts to a brute force attack in which all possibilities are tried. The attack works because users often choose easy-to-guess passwords. In anti-spam terminology this also refers to using a dictionary list to guess email account names in a spam campaign.

8.10 directory harvest attack

A technique to build a spam mailing list by sending all possible alphanumeric combinations for the user name to an email domain and building a database of all addresses that do not generate a reply. The harvesting software uses either a brute force approach (sending all possible alphanumeric combinations for the username), or a more selective method (for example, using all possible first initials followed by common surnames). In either case, the email server generally returns a "Not found" reply for all messages sent to a nonexistent address and none for those sent to valid addresses. The harvesting program builds a database of all addresses that do not generate a reply.

8.11 DNS A Records

A mapping of hostnames to IP addresses.

8.12 DNSBL

A published list of the IP addresses of known spam sources that is used as one factor in deciding to identify an email as spam. DNSBLs are chiefly used to publish lists of addresses linked to spamming. Most [MTA](#) software can be configured to reject or flag messages that have been sent from a site listed on one or more such lists.

8.13 DNS MX Records

DNS Mail Exchange Record map domain names to a list of mail exchange servers for that domain.

8.14 edge server

A server that operates on the security edge of an organization's intranet. That is, a computer running server applications that interact with systems outside of the control of the organization. In the email and anti-spam world, an MTA that sends and receives messages to and from other organization's MTAs.

8.15 email header

The meta information prepended to email messages to provide content information and log the message's path from sender to recipient. Headers are added by the sender's email client, and by each mail relay that passes the message toward its destination. Normally, the node that retrieves the message doesn't add any further headers. Typical email headers include: To, From, Subject, Date, User-Agent, and Received.

8.16 End User Web Interface (EUWI)

A web-based interface for end users that allows them to manage their PureMessage user-specific options. End users can manage blocked (quarantined) messages, modify their Allowed Senders and Blocked Senders lists, and configure other user-specific options.

8.17 false negative

A determination that an email message is not spam when in fact it is or that there are no viruses present when in fact there are.

8.18 false positive

A determination that a virus has been found or that an email message is spam when this is not the case.

8.19 glob

A minimal regular expression syntax used in searching.

8.20 Groups Web Interface

A web-based interface that allows a global administrator to delegate administrative responsibilities to group administrators based on groups/domains and/or roles. Tasks are delegated by way of access rights and can include quarantine management, reporting, list management and the configuration of certain policy settings. Group administrators can only access tabs and features that have been made available by the global administrator.

8.21 group

List of users to which policy settings can be applied. Groups determine which filtering actions are performed for which users.

8.22 heuristics

Anti-spam rules that provide general indications that a message is likely to contain spam or viruses. Each indicator contributes to the overall spam weight. Heuristics allow PureMessage to detect variants of certain kinds of spam and viruses, even if the specific variants have yet to be analyzed by SophosLabs.

8.23 internal hosts

Hosts that reside within your network, behind the gateway or proxy server.

8.24 Manager

A web-based graphical user interface (GUI) for the PureMessage system, as opposed to the command-line interface.

8.25 milter

An extension to sendmail (MTA) that allows administrators to add mail filters for filtering spam, viruses or applying mail policy.

8.26 MTA

A Mail Transfer Agent (MTA) is a service that transfers messages from the sender or another relay toward its destination. Often referred to as a mail relay or a mail hub.

8.27 network mask

Specifies the subnetwork and host parts of an IP address. Also known as a subnet mask, netmask or address mask. Network masks are usually represented in either dotted quad notation (for example, 255.255.255.0) or CIDR notation (for example, 192.0.2.0/24).

8.28 packages

Files that contain modularized grouped components used for software installation and upgrade.

8.29 phishing

Attempting to fraudulently acquire sensitive information, such as passwords and credit card details, by masquerading as a trustworthy person or business in an apparently official electronic communication, such as an email or an instant message. Also known as “carding” and “spoofing”.

8.30 policy

A set of rules and actions that specify criteria and behaviors when the criteria are met. As passes through the policy engine, it is tested to see if it meets specified criteria. When the traffic meet the criteria, an action is performed on it.

8.31 Postfix

An open source mail transfer agent (a computer program for the routing and delivery of email) that is intended as a fast, easy-to-administer, and secure alternative to the widely-used sendmail.

8.32 PostgreSQL

A free, object-relational database server (database management system), released under a flexible BSD-style license. It offers an alternative to other open-source database systems, as well as to proprietary systems. PostgreSQL is used by PureMessage to store report data, message quarantines, and end-user resources.

8.33 quarantine

A store of messages that cannot be delivered because they have been blocked by policy rules.

8.34 roles

Predetermined groupings of network services that can be divided between one or more systems on a network.

8.35 Scheduler

The PureMessage replacement for `cron`, the Unix/Linux command scheduler. Scheduler provides the advantage of being managed by PureMessage, which means that running the `pmx` command to start or stop PureMessage will also start and stop PureMessage’s scheduled jobs.

8.36 Sendmail

An open-source mail transfer agent (MTA), which routes and delivers email.

8.37 Sieve

A language for filtering mail at the time of final delivery.

8.38 SMTP

The Simple Mail Transfer Protocol (SMTP) is the standard protocol for email transmission across the internet.

8.39 SophosLabs

A network of skilled analysts who respond to evolving security threats such as viruses, spam, phishing schemes, spyware and other malware. Sophos Labs uses state-of-the-art big-data analytics to efficiently process the millions of emails, URLs, files, and other data points that come into the labs each day. Analysts are on duty to respond to new threats and analyze customer submissions 24/7/365.

8.40 spam score

The score assigned to a message by the anti-spam engine that indicates the relative likelihood that the message is spam. Anti-spam rules consist of a test definition and a "weight". If the test matches the message, the corresponding weight is added to the message's total spam score. Generally, multiple rules must be triggered by a message in order to result in a spam score high enough for an action to be taken. SophosLabs constantly analyzes emerging spam techniques and updates the ES4000 and PureMessage anti-spam rule sets accordingly.

8.41 spam

Unsolicited email, often sent to millions of recipients at a time. Spammers harvest recipient addresses from Usenet postings and web pages, obtain them from databases, or simply guess them by using common names and domains. Sending spam violates the Acceptable Use Policy (AUP) of most ISPs, and can lead to the termination of the sender's account. Many jurisdictions now consider spamming a crime, such as the US, which regulates via the CAN-SPAM Act of 2003.

8.42 SXL

The infrastructure that Sophos uses to submit real-time, DNS-based queries to SophosLabs regarding IP addresses, URLs within messages, and image fingerprints. Queries are triggered when the anti-spam engine has been unable to determine if a message is spam. These real-time lookups are enabled

by default and provide minimal latency between the time that Sophos makes new anti-spam data available and when it is available for use by the anti-spam engine.

8.43 TLS

Transport Layer Security (TLS) is a protocol that ensures privacy between communicating applications and their users on the Internet. When a server and client communicate, TLS ensures that no third party may eavesdrop or tamper with any message. TLS is the successor to the Secure Sockets Layer (SSL).

8.44 whitelist

A list that identifies addresses, hosts or IP addresses from which email will always be allowed without further processing. This type of list is also known as an “allow list”.