

NAME

qmgr - Postfix queue manager

SYNOPSIS

qmgr [generic Postfix daemon options]

DESCRIPTION

The **qmgr(8)** daemon awaits the arrival of incoming mail and arranges for its delivery via Postfix delivery processes. The actual mail routing strategy is delegated to the **trivial-rewrite(8)** daemon. This program expects to be run from the **master(8)** process manager.

Mail addressed to the local **double-bounce** address is logged and discarded. This stops potential loops caused by undeliverable bounce notifications.

MAIL QUEUES

The **qmgr(8)** daemon maintains the following queues:

incoming

Inbound mail from the network, or mail picked up by the local **pickup(8)** daemon from the **mail-drop** directory.

active Messages that the queue manager has opened for delivery. Only a limited number of messages is allowed to enter the **active** queue (leaky bucket strategy, for a fixed delivery rate).

deferred

Mail that could not be delivered upon the first attempt. The queue manager implements exponential backoff by doubling the time between delivery attempts.

corrupt

Unreadable or damaged queue files are moved here for inspection.

hold Messages that are kept "on hold" are kept here until someone sets them free.

DELIVERY STATUS REPORTS

The **qmgr(8)** daemon keeps an eye on per-message delivery status reports in the following directories. Each status report file has the same name as the corresponding message file:

bounce Per-recipient status information about why mail is bounced. These files are maintained by the **bounce(8)** daemon.

defer Per-recipient status information about why mail is delayed. These files are maintained by the **defer(8)** daemon.

trace Per-recipient status information as requested with the Postfix "**sendmail -v**" or "**sendmail -bv**" command. These files are maintained by the **trace(8)** daemon.

The **qmgr(8)** daemon is responsible for asking the **bounce(8)**, **defer(8)** or **trace(8)** daemons to send delivery reports.

STRATEGIES

The queue manager implements a variety of strategies for either opening queue files (input) or for message delivery (output).

leaky bucket

This strategy limits the number of messages in the **active** queue and prevents the queue manager from running out of memory under heavy load.

fairness

When the **active** queue has room, the queue manager takes one message from the **incoming** queue and one from the **deferred** queue. This prevents a large mail backlog from blocking the delivery of new mail.

slow start

This strategy eliminates "thundering herd" problems by slowly adjusting the number of parallel deliveries to the same destination.

round robin

The queue manager sorts delivery requests by destination. Round-robin selection prevents one destination from dominating deliveries to other destinations.

exponential backoff

Mail that cannot be delivered upon the first attempt is deferred. The time interval between delivery attempts is doubled after each attempt.

destination status cache

The queue manager avoids unnecessary delivery attempts by maintaining a short-term, in-memory list of unreachable destinations.

preemptive message scheduling

The queue manager attempts to minimize the average per-recipient delay while still preserving the correct per-message delays, using a sophisticated preemptive message scheduling.

TRIGGERS

On an idle system, the queue manager waits for the arrival of trigger events, or it waits for a timer to go off. A trigger is a one-byte message. Depending on the message received, the queue manager performs one of the following actions (the message is followed by the symbolic constant used internally by the software):

D (QMGR_REQ_SCAN_DEFERRED)

Start a deferred queue scan. If a deferred queue scan is already in progress, that scan will be restarted as soon as it finishes.

I (QMGR_REQ_SCAN_INCOMING)

Start an incoming queue scan. If an incoming queue scan is already in progress, that scan will be restarted as soon as it finishes.

A (QMGR_REQ_SCAN_ALL)

Ignore deferred queue file time stamps. The request affects the next deferred queue scan.

F (QMGR_REQ_FLUSH_DEAD)

Purge all information about dead transports and destinations.

W (TRIGGER_REQ_WAKEUP)

Wakeup call, This is used by the master server to instantiate servers that should not go away forever. The action is to start an incoming queue scan.

The **qmgr(8)** daemon reads an entire buffer worth of triggers. Multiple identical trigger requests are collapsed into one, and trigger requests are sorted so that **A** and **F** precede **D** and **I**. Thus, in order to force a deferred queue run, one would request **A F D**; in order to notify the queue manager of the arrival of new mail one would request **I**.

STANDARDS

RFC 3463 (Enhanced status codes)

RFC 3464 (Delivery status notifications)

SECURITY

The **qmgr(8)** daemon is not security sensitive. It reads single-character messages from untrusted local users, and thus may be susceptible to denial of service attacks. The **qmgr(8)** daemon does not talk to the outside world, and it can be run at fixed low privilege in a chrooted environment.

DIAGNOSTICS

Problems and transactions are logged to the syslog daemon. Corrupted message files are saved to the **corrupt** queue for further inspection.

Depending on the setting of the **notify_classes** parameter, the postmaster is notified of bounces and of other trouble.

BUGS

A single queue manager process has to compete for disk access with multiple front-end processes such as **cleanup(8)**. A sudden burst of inbound mail can negatively impact outbound delivery rates.

CONFIGURATION PARAMETERS

Changes to **main.cf** are not picked up automatically as **qmgr(8)** is a persistent process. Use the "**postfix reload**" command after a configuration change.

The text below provides only a parameter summary. See **postconf(5)** for more details including examples.

In the text below, *transport* is the first field in a **master.cf** entry.

COMPATIBILITY CONTROLS

Available before Postfix version 2.5:

allow_min_user (no)

Allow a sender or recipient address to have '-' as the first character.

Available with Postfix version 2.7 and later:

default_filter_nexthop (empty)

When a content_filter or FILTER request specifies no explicit next-hop destination, use \$default_filter_nexthop instead; when that value is empty, use the domain in the recipient address.

ACTIVE QUEUE CONTROLS

qmgr_clog_warn_time (300s)

The minimal delay between warnings that a specific destination is clogging up the Postfix active queue.

qmgr_message_active_limit (20000)

The maximal number of messages in the active queue.

qmgr_message_recipient_limit (20000)

The maximal number of recipients held in memory by the Postfix queue manager, and the maximal size of the short-term, in-memory "dead" destination status cache.

qmgr_message_recipient_minimum (10)

The minimal number of in-memory recipients for any message.

default_recipient_limit (20000)

The default per-transport upper limit on the number of in-memory recipients.

transport_recipient_limit (\$default_recipient_limit)

Idem, for delivery via the named message *transport*.

default_extra_recipient_limit (1000)

The default value for the extra per-transport limit imposed on the number of in-memory recipients.

transport_extra_recipient_limit (\$default_extra_recipient_limit)

Idem, for delivery via the named message *transport*.

Available in Postfix version 2.4 and later:

default_recipient_refill_limit (100)

The default per-transport limit on the number of recipients refilled at once.

transport_recipient_refill_limit (\$default_recipient_refill_limit)

Idem, for delivery via the named message *transport*.

default_recipient_refill_delay (5s)

The default per-transport maximum delay between recipients refills.

transport_recipient_refill_delay (\$default_recipient_refill_delay)

Idem, for delivery via the named message *transport*.

DELIVERY CONCURRENCY CONTROLS

initial_destination_concurrency (5)

The initial per-destination concurrency level for parallel delivery to the same destination.

default_destination_concurrency_limit (20)

The default maximal number of parallel deliveries to the same destination.

transport_destination_concurrency_limit (\$default_destination_concurrency_limit)

Idem, for delivery via the named message *transport*.

Available in Postfix version 2.5 and later:

transport_initial_destination_concurrency (\$initial_destination_concurrency)

Initial concurrency for delivery via the named message *transport*.

default_destination_concurrency_failed_cohort_limit (1)

How many pseudo-cohorts must suffer connection or handshake failure before a specific destination is considered unavailable (and further delivery is suspended).

transport_destination_concurrency_failed_cohort_limit (\$default_destination_concurrency_failed_cohort_limit)

Idem, for delivery via the named message *transport*.

default_destination_concurrency_negative_feedback (1)

The per-destination amount of delivery concurrency negative feedback, after a delivery completes with a connection or handshake failure.

transport_destination_concurrency_negative_feedback (\$default_destination_concurrency_negative_feedback)

Idem, for delivery via the named message *transport*.

default_destination_concurrency_positive_feedback (1)

The per-destination amount of delivery concurrency positive feedback, after a delivery completes without connection or handshake failure.

transport_destination_concurrency_positive_feedback (\$default_destination_concurrency_positive_feedback)

Idem, for delivery via the named message *transport*.

destination_concurrency_feedback_debug (no)

Make the queue manager's feedback algorithm verbose for performance analysis purposes.

RECIPIENT SCHEDULING CONTROLS**default_destination_recipient_limit (50)**

The default maximal number of recipients per message delivery.

transport_destination_recipient_limit (\$default_destination_recipient_limit)

Idem, for delivery via the named message *transport*.

MESSAGE SCHEDULING CONTROLS**default_delivery_slot_cost (5)**

How often the Postfix queue manager's scheduler is allowed to preempt delivery of one message with another.

transport_delivery_slot_cost (\$default_delivery_slot_cost)

Idem, for delivery via the named message *transport*.

default_minimum_delivery_slots (3)

How many recipients a message must have in order to invoke the Postfix queue manager's scheduling algorithm at all.

transport_minimum_delivery_slots (\$default_minimum_delivery_slots)

Idem, for delivery via the named message *transport*.

default_delivery_slot_discount (50)

The default value for transport-specific `_delivery_slot_discount` settings.

***transport_delivery_slot_discount* (\$default_delivery_slot_discount)**

Idem, for delivery via the named message *transport*.

default_delivery_slot_loan (3)

The default value for transport-specific *_delivery_slot_loan* settings.

***transport_delivery_slot_loan* (\$default_delivery_slot_loan)**

Idem, for delivery via the named message *transport*.

OTHER RESOURCE AND RATE CONTROLS

minimal_backoff_time (300s)

The minimal time between attempts to deliver a deferred message; prior to Postfix 2.4 the default value was 1000s.

maximal_backoff_time (4000s)

The maximal time between attempts to deliver a deferred message.

maximal_queue_lifetime (5d)

Consider a message as undeliverable, when delivery fails with a temporary error, and the time in the queue has reached the *maximal_queue_lifetime* limit.

queue_run_delay (300s)

The time between deferred queue scans by the queue manager; prior to Postfix 2.4 the default value was 1000s.

transport_retry_time (60s)

The time between attempts by the Postfix queue manager to contact a malfunctioning message delivery transport.

Available in Postfix version 2.1 and later:

bounce_queue_lifetime (5d)

Consider a bounce message as undeliverable, when delivery fails with a temporary error, and the time in the queue has reached the *bounce_queue_lifetime* limit.

Available in Postfix version 2.5 and later:

default_destination_rate_delay (0s)

The default amount of delay that is inserted between individual deliveries to the same destination; the resulting behavior depends on the value of the corresponding per-destination recipient limit.

***transport_destination_rate_delay* \$default_destination_rate_delay**

Idem, for delivery via the named message *transport*.

Available in Postfix version 3.1 and later:

default_transport_rate_delay (0s)

The default amount of delay that is inserted between individual deliveries over the same message delivery transport, regardless of destination.

***transport_transport_rate_delay* \$default_transport_rate_delay**

Idem, for delivery via the named message *transport*.

SAFETY CONTROLS

qmgr_daemon_timeout (1000s)

How much time a Postfix queue manager process may take to handle a request before it is terminated by a built-in watchdog timer.

qmgr_ipc_timeout (60s)

The time limit for the queue manager to send or receive information over an internal communication channel.

Available in Postfix version 3.1 and later:

address_verify_pending_request_limit (see **'postconf -d' output**)

A safety limit that prevents address verification requests from overwhelming the Postfix queue.

MISCELLANEOUS CONTROLS**config_directory** (see **'postconf -d' output**)

The default location of the Postfix main.cf and master.cf configuration files.

defer_transports (empty)

The names of message delivery transports that should not deliver mail unless someone issues **"sendmail -q"** or equivalent.

delay_logging_resolution_limit (2)

The maximal number of digits after the decimal point when logging sub-second delay values.

helpful_warnings (yes)

Log warnings about problematic configuration settings, and provide helpful suggestions.

process_id (read-only)

The process ID of a Postfix command or daemon process.

process_name (read-only)

The process name of a Postfix command or daemon process.

queue_directory (see **'postconf -d' output**)

The location of the Postfix top-level queue directory.

syslog_facility (mail)

The syslog facility of Postfix logging.

syslog_name (see **'postconf -d' output**)

The mail system name that is prepended to the process name in syslog records, so that **"smtpd"** becomes, for example, **"postfix/smtpd"**.

Available in Postfix version 3.0 and later:

confirm_delay_cleared (no)

After sending a "your message is delayed" notification, inform the sender when the delay clears up.

FILES

/var/spool/postfix/incoming, incoming queue
 /var/spool/postfix/active, active queue
 /var/spool/postfix/deferred, deferred queue
 /var/spool/postfix/bounce, non-delivery status
 /var/spool/postfix/defer, non-delivery status
 /var/spool/postfix/trace, delivery status

SEE ALSO

[trivial-rewrite\(8\)](#),
 address routing
[bounce\(8\)](#),
 delivery status reports
[postconf\(5\)](#),
 configuration parameters
[master\(5\)](#),
 generic daemon options
[master\(8\)](#),
 process manager
[syslogd\(8\)](#), system logging

README FILES

Use **"postconf readme_directory"** or **"postconf html_directory"** to locate this information.
 SCHEDULER_README, scheduling algorithm

QSHAPE_README, Postfix queue analysis

LICENSE

The Secure Mailer license must be distributed with this software.

AUTHOR(S)

Wietse Venema
IBM T.J. Watson Research
P.O. Box 704
Yorktown Heights, NY 10598, USA

Preemptive scheduler enhancements:

Patrik Rak
Modra 6
155 00, Prague, Czech Republic

Wietse Venema
Google, Inc.
111 8th Avenue
New York, NY 10011, USA