

NAME

master – Postfix master process configuration file format

DESCRIPTION

The Postfix mail system is implemented by small number of (mostly) client commands that are invoked by users, and by a larger number of services that run in the background.

Postfix services are implemented by daemon processes. These run in the background under control of the **master**(8) process. The master.cf configuration file defines how a client program connects to a service, and what daemon program runs when a service is requested. Most daemon processes are short-lived and terminate voluntarily after serving **max_use** clients, or after inactivity for **max_idle** or more units of time.

All daemons specified here must speak a Postfix-internal protocol. In order to execute non-Postfix software use the **local**(8), **pipe**(8) or **spawn**(8) services, or run the server under control by **inetd**(8) or equivalent.

After changing master.cf you must execute "**postfix reload**" to reload the configuration.

SYNTAX

The general format of the master.cf file is as follows:

- Empty lines and whitespace-only lines are ignored, as are lines whose first non-whitespace character is a '#'.
- A logical line starts with non-whitespace text. A line that starts with whitespace continues a logical line.
- Each logical line defines a single Postfix service. Each service is identified by its name and type as described below. When multiple lines specify the same service name and type, only the last one is remembered. Otherwise, the order of master.cf service definitions does not matter.

Each logical line consists of eight fields separated by whitespace. These are described below in the order as they appear in the master.cf file.

Where applicable a field of "-" requests that the built-in default value be used. For boolean fields specify "y" or "n" to override the default value.

Service name

The service name syntax depends on the service type as described next.

Service type

Specify one of the following service types:

inet The service listens on a TCP/IP socket and is accessible via the network.

The service name is specified as *host:port*, denoting the host and port on which new connections should be accepted. The host part (and colon) may be omitted. Either host or port may be given in symbolic form (host or service name) or in numeric form (IP address or port number). Host information may be enclosed inside "[]"; this form is necessary only with IPv6 addresses.

Examples: a service named **127.0.0.1:smtp** or **::1:smtp** receives mail via the loopback interface only; and a service named **10025** accepts connections on TCP port 10025 via all interfaces configured with the **inet_interfaces** parameter.

Note: with Postfix version 2.2 and later specify "**inet_interfaces = loopback-only**" in main.cf, instead of hard-coding loopback IP address information in master.cf or in main.cf.

unix The service listens on a UNIX-domain socket and is accessible for local clients only.

The service name is a pathname relative to the Postfix queue directory (pathname controlled with the **queue_directory** configuration parameter in main.cf).

On Solaris 8 and earlier systems the **unix** type is implemented with streams sockets.

fifo The service listens on a FIFO (named pipe) and is accessible for local clients only.

The service name is a pathname relative to the Postfix queue directory (pathname controlled with the **queue_directory** configuration parameter in main.cf).

pass The service listens on a UNIX-domain socket, and is accessible to local clients only. It receives one open connection (file descriptor passing) per connection request.

The service name is a pathname relative to the Postfix queue directory (pathname controlled with the **queue_directory** configuration parameter in main.cf).

On Solaris 8 and earlier systems the **pass** type is implemented with streams sockets.

This feature is available as of Postfix version 2.5.

Private (default: y)

Whether or not access is restricted to the mail system. Internet (type **inet**) services can't be private.

Unprivileged (default: y)

Whether the service runs with root privileges or as the owner of the Postfix system (the owner name is controlled by the **mail_owner** configuration variable in the main.cf file).

The **local(8)**, **pipe(8)**, **spawn(8)**, and **virtual(8)** daemons require privileges.

Chroot (default: Postfix >= 3.0: n, Postfix <3.0: y)

Whether or not the service runs chrooted to the mail queue directory (pathname is controlled by the **queue_directory** configuration variable in the main.cf file).

Chroot should not be used with the **local(8)**, **pipe(8)**, **spawn(8)**, and **virtual(8)** daemons. Although the **proxymap(8)** server can run chrooted, doing so defeats most of the purpose of having that service in the first place.

The files in the examples/chroot-setup subdirectory of the Postfix source archive show set up a Postfix chroot environment on a variety of systems. See also BASIC_CONFIGURATION_README for issues related to running daemons chrooted.

Wake up time (default: 0)

Automatically wake up the named service after the specified number of seconds. The wake up is implemented by connecting to the service and sending a wake up request. A ? at the end of the wake-up time field requests that no wake up events be sent before the first time a service is used. Specify 0 for no automatic wake up.

The **pickup(8)**, **qmgr(8)** and **flush(8)** daemons require a wake up timer.

Process limit (default: \$default_process_limit)

The maximum number of processes that may execute this service simultaneously. Specify 0 for no process count limit.

NOTE: Some Postfix services must be configured as a single-process service (for example, **qmgr(8)**) and some services must be configured with no process limit (for example, **cleanup(8)**). These limits must not be changed.

Command name + arguments

The command to be executed. Characters that are special to the shell such as ">" or "|" have no special meaning here, and quotes cannot be used to protect arguments containing whitespace. To protect whitespace, use "{" and "}" as described below.

The command name is relative to the Postfix daemon directory (pathname is controlled by the **daemon_directory** configuration variable).

The command argument syntax for specific commands is specified in the respective daemon manual page.

The following command-line options have the same effect for all daemon programs:

-D Run the daemon under control by the command specified with the **debugger_command** variable in the main.cf configuration file. See **DEBUG_README** for hints and tips.

-o { name = value } (long form, Postfix >= 3.0)

-o name=value (short form)

Override the named main.cf configuration parameter. The parameter value can refer to other parameters as *\$name* etc., just like in main.cf. See **postconf(5)** for syntax.

NOTE 1: With the "long form" shown above, whitespace after "{", around "=", and before "}" is ignored, and whitespace within the parameter value is preserved.

NOTE 2: with the "short form" shown above, do not specify whitespace around the "=" or in parameter values. To specify a parameter value that contains whitespace, use the long form described above, or use commas instead of spaces, or specify the value in main.cf. Example:

```
/etc/postfix/master.cf:
submission inet .... smtpd
    -o smtpd_xxx_yyy=$submission_xxx_yyy
```

```
/etc/postfix/main.cf
submission_xxx_yyy = text with whitespace...
```

NOTE 3: Over-zealous use of parameter overrides makes the Postfix configuration hard to understand and maintain. At a certain point, it might be easier to configure multiple instances of Postfix, instead of configuring multiple personalities via master.cf.

-v Increase the verbose logging level. Specify multiple **-v** options to make a Postfix daemon process increasingly verbose.

Other command-line arguments

Specify "{" and "}" around command arguments that contain whitespace (Postfix 3.0 and later). Whitespace after "{" and before "}" is ignored.

SEE ALSO

master(8), process manager
postconf(5), configuration parameters

README FILES

Use "**postconf readme_directory**" or "**postconf html_directory**" to locate this information.
BASIC_CONFIGURATION_README, basic configuration
DEBUG_README, Postfix debugging

LICENSE

The Secure Mailer license must be distributed with this software.

AUTHOR(S)

Initial version by
Magnus Baeck
Lund Institute of Technology
Sweden

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