

Section 13 - Starting the daemon and the use of network interfaces

13. Starting the daemon and the use of network interfaces

A host that is connected to a TCP/IP network may have one or more physical hardware network interfaces. Each of these interfaces may be configured as one or more "logical" interfaces, which are the entities that a program actually works with. Each of these logical interfaces is associated with an IP address. In addition, TCP/IP software supports "loopback" interfaces (127.0.0.1 in IPv4 and ::1 in IPv6), which do not use any physical hardware. Exim requires knowledge about the host's interfaces for use in three different circumstances:

-

When a listening daemon is started, Exim needs to know which interfaces and ports to listen on.

-

When Exim is routing an address, it needs to know which IP addresses are associated with local interfaces. This is required for the correct processing of MX lists by removing the local host and others with the same or higher priority values. Also, Exim needs to detect cases when an address is routed to an IP address that in fact belongs to the local host. Unless the self router option or the allow_localhost option of the smtp transport is set (as appropriate), this is treated as an error situation.

-

When Exim connects to a remote host, it may need to know which interface to use for the outgoing connection.

Exim's default behaviour is likely to be appropriate in the vast majority of cases. If your host has only one interface, and you want all its IP addresses to be treated in the same way, and you are using only the standard SMTP port, you should not need to take any special action. The rest of this chapter does not apply to you.

In a more complicated situation you may want to listen only on certain interfaces, or on different ports, and for this reason there are a number of options that can be used to influence Exim's behaviour. The rest of this chapter describes how they operate.

When a message is received over TCP/IP, the interface and port that were actually used are set in \$interface_address and \$interface_port.

13.1 Starting a listening daemon

When a listening daemon is started (by means of the -bd command line option), the interfaces and ports on which it listens are controlled by the following options:

-

daemon_smtp_ports contains a list of default ports. (For backward compatibility, this option can also be specified in the singular.)

-

local_interfaces contains list of interface IP addresses on which to listen. Each item may optionally also specify a port.

The default list separator in both cases is a colon, but this can be changed as described in section 6.19. When IPv6 addresses are involved, it is usually best to change the separator to avoid having to double all the colons. For example:

```
local_interfaces = <; 127.0.0.1 ; \
    192.168.23.65 ; \
    ::1 ; \
    3ffe:ffff:836f::fe86:a061
```

There are two different formats for specifying a port along with an IP address in local_interfaces:

-

The port is added onto the address with a dot separator. For example, to listen on port 1234 on two different IP addresses: local_interfaces = <; 192.168.23.65.1234 ; \

```
    3ffe:ffff:836f::fe86:a061.1234
```

-

The IP address is enclosed in square brackets, and the port is added with a colon separator, for example:

```
local_interfaces = <; [192.168.23.65]:1234 ; \
    [3ffe:ffff:836f::fe86:a061]:1234
```

When a port is not specified, the value of `daemon_smtp_ports` is used. The default setting contains just one port:
`daemon_smtp_ports = smtp`

If more than one port is listed, each interface that does not have its own port specified listens on all of them. Ports that are listed in `daemon_smtp_ports` can be identified either by name (defined in `/etc/services`) or by number. However, when ports are given with individual IP addresses in `local_interfaces`, only numbers (not names) can be used.

13.2 Special IP listening addresses

The addresses `0.0.0.0` and `::0` are treated specially. They are interpreted as "all IPv4 interfaces" and "all IPv6 interfaces", respectively. In each case, Exim tells the TCP/IP stack to "listen on all IPvx interfaces" instead of setting up separate listening sockets for each interface. The default value of `local_interfaces` is `local_interfaces = 0.0.0.0`

when Exim is built without IPv6 support; otherwise it is: `local_interfaces = <; ::0 ; 0.0.0.0`

Thus, by default, Exim listens on all available interfaces, on the SMTP port. 13.3 Overriding `local_interfaces` and `daemon_smtp_ports`

The `-oX` command line option can be used to override the values of `daemon_smtp_ports` and/or `local_interfaces` for a particular daemon instance. Another way of doing this would be to use macros and the `-D` option. However, `-oX` can be used by any admin user, whereas modification of the runtime configuration by `-D` is allowed only when the caller is root or `exim`.

The value of `-oX` is a list of items. The default colon separator can be changed in the usual way if required. If there are any items that do not contain dots or colons (that is, are not IP addresses), the value of `daemon_smtp_ports` is replaced by the list of those items. If there are any items that do contain dots or colons, the value of `local_interfaces` is replaced by those items. Thus, for example, `-oX 1225`

overrides `daemon_smtp_ports`, but leaves `local_interfaces` unchanged, whereas `-oX 192.168.34.5.1125`

overrides `local_interfaces`, leaving `daemon_smtp_ports` unchanged. (However, since `local_interfaces` now contains no items without ports, the value of `daemon_smtp_ports` is no longer relevant in this example.) 13.4 Support for the obsolete SSMTP (or SMTPS) protocol

Exim supports the obsolete SSMTP protocol (also known as SMTPS) that was used before the STARTTLS command was standardized for SMTP. Some legacy clients still use this protocol. If the `tls_on_connect_ports` option is set to a list of port numbers, connections to those ports must use SSMTP. The most common use of this option is expected to be `tls_on_connect_ports = 465`

because 465 is the usual port number used by the legacy clients. There is also a command line option `-tls-on-connect`, which forces all ports to behave in this way when a daemon is started.

Warning: Setting `tls_on_connect_ports` does not of itself cause the daemon to listen on those ports. You must still specify them in `daemon_smtp_ports`, `local_interfaces`, or the `-oX` option. (This is because `tls_on_connect_ports` applies to `inetd` connections as well as to connections via the daemon.) 13.5 IPv6 address scopes

IPv6 addresses have "scopes", and a host with multiple hardware interfaces can, in principle, have the same link-local IPv6 address on different interfaces. Thus, additional information is needed, over and above the IP address, to distinguish individual interfaces. A convention of using a percent sign followed by something (often the interface name) has been adopted in some cases, leading to addresses like this: `fe80::202:b3ff:fe03:45c1%eth0`

To accommodate this usage, a percent sign followed by an arbitrary string is allowed at the end of an IPv6 address. By default, Exim calls `getaddrinfo()` to convert a textual IPv6 address for actual use. This function recognizes the percent convention in operating systems that support it, and it processes the address appropriately. Unfortunately, some older libraries have problems with `getaddrinfo()`. If `IPV6_USE_INET_PTON=yes`

is set in Local/Makefile (or an OS-dependent Makefile) when Exim is built, Exim uses `inet_pton()` to convert a textual IPv6 address for actual use, instead of `getaddrinfo()`. (Before version 4.14, it always used this function.) Of course, this means that the additional functionality of `getaddrinfo()` – recognizing scoped addresses – is lost.

13.6 Disabling IPv6

Sometimes it happens that an Exim binary that was compiled with IPv6 support is run on a host whose kernel does not support IPv6. The binary will fall back to using IPv4, but it may waste resources looking up AAAA records, and trying to connect to IPv6 addresses, causing delays to mail delivery. If you set the `disable_ipv6` option true, even if the Exim binary has IPv6 support, no IPv6 activities take place. AAAA records are never looked up, and any IPv6 addresses that are listed in `local_interfaces`, data for the manualroute router, etc. are ignored. If IP literals are enabled, the `ipliteral` router declines to handle IPv6 literal addresses.

On the other hand, when IPv6 is in use, there may be times when you want to disable it for certain hosts or domains. You can use the `dns_ipv4_lookup` option to globally suppress the lookup of AAAA records for specified domains, and you can use the `ignore_target_hosts` generic router option to ignore IPv6 addresses in an individual router. 13.7 Examples of starting a listening daemon

The default case in an IPv6 environment is `daemon_smtp_ports = smtp`
`local_interfaces = <; ::0 ; 0.0.0.0`

This specifies listening on the `smtp` port on all IPv6 and IPv4 interfaces. Either one or two sockets may be used, depending on the characteristics of the TCP/IP stack. (This is complicated and messy; for more information, read the comments in the `daemon.c` source file.)

To specify listening on ports 25 and 26 on all interfaces: `daemon_smtp_ports = 25 : 26`

(leaving `local_interfaces` at the default setting) or, more explicitly: `local_interfaces = <; ::0.25 ; ::0.26 \`
`0.0.0.0.25 ; 0.0.0.0.26`

To listen on the default port on all IPv4 interfaces, and on port 26 on the IPv4 loopback address only: `local_interfaces = 0.0.0.0 : 127.0.0.1.26`

To specify listening on the default port on specific interfaces only: `local_interfaces = 192.168.34.67 : 192.168.34.67`

Warning: Such a setting excludes listening on the loopback interfaces. 13.8 Recognising the local host

The `local_interfaces` option is also used when Exim needs to determine whether or not an IP address refers to the local host. That is, the IP addresses of all the interfaces on which a daemon is listening are always treated as local.

For this usage, port numbers in `local_interfaces` are ignored. If either of the items `0.0.0.0` or `::0` are encountered, Exim gets a complete list of available interfaces from the operating system, and extracts the relevant (that is, IPv4 or IPv6) addresses to use for checking.

Some systems set up large numbers of virtual interfaces in order to provide many virtual web servers. In this situation, you may want to listen for email on only a few of the available interfaces, but nevertheless treat all interfaces as local when routing. You can do this by setting `extra_local_interfaces` to a list of IP addresses, possibly including the “all” wildcard values. These addresses are recognized as local, but are not used for listening. Consider this example: `local_interfaces = <; 127.0.0.1 ; ::1 ; \`
`192.168.53.235 ; \`
`3ffe:2101:12:1:a00:20ff:fe86:a061`

`extra_local_interfaces = <; ::0 ; 0.0.0.0`

The daemon listens on the loopback interfaces and just one IPv4 and one IPv6 address, but all available interface addresses are treated as local when Exim is routing.

In some environments the local host name may be in an MX list, but with an IP address that is not assigned to any local interface. In other cases it may be desirable to treat other host names as if they referred to the local host. Both these

cases can be handled by setting the `hosts_treat_as_local` option. This contains host names rather than IP addresses. When a host is referenced during routing, either via an MX record or directly, it is treated as the local host if its name matches `hosts_treat_as_local`, or if any of its IP addresses match `local_interfaces` or `extra_local_interfaces`.

13.9 Delivering to a remote host

Delivery to a remote host is handled by the `smtp` transport. By default, it allows the system's TCP/IP functions to choose which interface to use (if there is more than one) when connecting to a remote host. However, the `interface` option can be set to specify which interface is used. See the description of the `smtp` transport in chapter 30 for more details.