Binding to Addresses and Ports

Configuring Apache HTTP Server to listen on specific addresses and ports.

Overview

Related Modules	Related Directives
core	<VirtualHost>
mpm_common	Listen

When httpd starts, it binds to some port and address on the local machine and waits for incoming requests. By default, it listens to all addresses on the machine. However, it may need to be told to listen on specific ports, or only on selected addresses, or a combination of both. This is often combined with the Virtual Host feature, which determines how `httpd` responds to different IP addresses, hostnames and ports.

The `Listen` directive tells the server to accept incoming requests only on the specified port(s) or address-and-port combinations. If only a port number is specified in the `Listen` directive, the server listens to the given port on all interfaces. If an IP address is given as well as a port, the server will listen on the given port and interface. Multiple `Listen` directives may be used to specify a number of addresses and ports to listen on. The server will respond to requests from any of the listed addresses and ports.

For example, to make the server accept connections on both port 80 and port 8000, on all interfaces, use:

```
Listen 80
Listen 8000
```

To make the server accept connections on port 80 for one interface, and port 8000 on another, use

```
Listen 192.0.2.1:80
Listen 192.0.2.5:8000
```

IPv6 addresses must be enclosed in square brackets, as in the following example:

```
Listen [2001:db8::a00:20ff:fea7:ccea]:80
```

Overlapping `Listen` directives will result in a fatal error which will prevent the server from starting up.

```
(48)Address already in use: make_sock: could not
bind to address [::]:80
```

See the discussion in the wiki for further troubleshooting tips.

Changing Listen configuration on restart

When httpd is restarted, special consideration must be made for changes to `Listen` directives. During a restart, httpd keeps ports bound (as in the original configuration) to avoid generating "Connection refused" errors for any new attempts to connect to the server. If changes are made to the set of `Listen`

directives used which conflict with the old configuration, configuration will fail and the server will terminate.

For example, changing from configuration:

```
Listen 127.0.0.1:80
```

to the following may fail, because binding to port 80 across all addresses conflicts with binding to port 80 on just 127.0.0.1.

```
Listen 80
```

To have such configuration changes take effect, it is necessary to stop and then start the server.

Special IPv6 Considerations

A growing number of platforms implement IPv6, and APR supports IPv6 on most of these platforms, allowing httpd to allocate IPv6 sockets, and to handle requests sent over IPv6.

One complicating factor for httpd administrators is whether or not an IPv6 socket can handle both IPv4 connections and IPv6 connections. Handling IPv4 connections with an IPv6 socket uses IPv4-mapped IPv6 addresses, which are allowed by default on most platforms, but are disallowed by default on FreeBSD, NetBSD, and OpenBSD, in order to match the system-wide policy on those platforms. On systems where it is disallowed by default, a special `configure` parameter can change this behavior for httpd.

On the other hand, on some platforms, such as Linux and Tru64, the **only** way to handle both IPv6 and IPv4 is to use mapped addresses. If you want `httpd` to handle IPv4 and IPv6 connections with a minimum of sockets, which requires using IPv4-mapped IPv6 addresses, specify the `--enable-v4-mapped` `configure` option.

`--enable-v4-mapped` is the default on all platforms except FreeBSD, NetBSD, and OpenBSD, so this is probably how your httpd was built.

If you want httpd to handle IPv4 connections only, regardless of what your platform and APR will support, specify an IPv4 address on all `Listen` directives, as in the following examples:

```
Listen 0.0.0.0:80
Listen 192.0.2.1:80
```

If your platform supports it and you want httpd to handle IPv4 and IPv6 connections on separate sockets (i.e., to disable IPv4-mapped addresses), specify the `--disable-v4-mapped` `configure` option. `--disable-v4-mapped` is the default on FreeBSD, NetBSD, and OpenBSD.

Specifying the protocol with Listen

The optional second *protocol* argument of `Listen` is not required for most configurations. If not specified, `https` is the default for port 443 and `http` the default for all other ports. The protocol is used to determine which module should handle a request, and to apply protocol specific optimizations with the `AcceptFilter` directive.

You only need to set the protocol if you are running on non-standard ports. For example, running an `https` site on port 8443:

```
Listen 192.170.2.1:8443 https
```

How This Works With Virtual Hosts

The `Listen` directive does not implement Virtual Hosts - it only tells the main server what addresses and ports to listen on. If no `<VirtualHost>` directives are used, the server will behave in the same way for all accepted requests. However, `<VirtualHost>` can be used to specify a different behavior for one or more of the addresses or ports. To implement a VirtualHost, the server must first be told to listen to the address and port to be used. Then a `<VirtualHost>` section should be created for the specified address and port to set the behavior of this virtual host. Note that if the `<VirtualHost>` is set for an address and port that the server is not listening to, it cannot be accessed.