

NETWORKING – HOW TO TEST HTTP (PORT 80) USING TELNET

Telnet is a long-time favorite for troubleshooting connection issues. It can be used to attempt to connect to a server on a given port. Generally you can quickly ascertain whether sometime is "listening" on the remote side--which will confirm that

- (1) you can get to the remote [qs server] from your test box, and...
- (2) that a server is running and bound to the specified port you requested.

More specifically, telnet can be used to test **HTTP** scenarios, effectively stripping through all the friendly layers a browser usually hides from web visitor eyes. Troubleshooting HTTP using telnet is a great way to review exactly what is getting returned to your browser during a web request for a particular **url**. This allows you to see headers and collect other information related to the web server.

Here is a simple example making an **HTTP 1.1** request. First, get to a command prompt. For example, if you are using **Windows, Start -> Run -> cmd**. (Though if you are using Windows, you may want to use something like **CYGWIN, or Putty** rather than the DOS prompt).

INPUT:

```
>telnet www.esqsoft.globalservers.com 80
```

Note: this launches **telnet** and tells it to look up and connect to the server for "www.esqsoft.globalservers.com" on port 80.

OUTPUT:

```
Trying 64.136.56.122... Connected to www.esqsoft.globalservers.com.  
Escape character is '^]'.
```

INPUT:

```
GET /index.htm HTTP/1.1 host: www.esqsoft.globalservers.com
```

Notes:

- 1) no OS cursor caret, the connection is open so you just get a blank line waiting for input.
- 2) need to hit <enter> twice after the "host:"

OUTPUT:

HTTP/1.1 200 OK

Date: Mon, 18 Apr 2005 16:38:00 GMT

Server: Apache/1.3.27 (Unix) (Red-Hat/Linux) mod_ssl/2.8.12 OpenSSL/0.9.6
PHP/4.1.2 mod_perl/1.27 FrontPage/5.0.2.2623

Last-Modified: Thu, 01 Jul 2004 01:16:05 GMT

ETag: "158e008c-182c-40e365d5"

Accept-Ranges: bytes

Content-Length: 6188

Connection: close

Content-Type: text/html