IIS 7 and the Web.config File

Posted At : January 6, 2011 6:42 PM | Posted By : Mark Kruger
Related Categories: Hosting and Networking

If you are new to IIS 7 you may not know about the Web.config file. This file acquires its initial properties from the global settings that you set at the server level (as opposed to the site level). If you make certain changes to the global settings (like adding a default doc for example) then a new web.config file is automatically created and put in the root of each site you add. Or possibly it's created when you fiddle with the site specific settings and deviate from the global settings. I'm not clear on when it *is* and *is not* created. But you can of course create one for yourself. The format looks like this:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
<system.webServer>
<defaultDocument>
<files>
<clear />
<add value="index.cfm" />
<add value="index.html" />
<add value="Default.htm" />
<add value="Default.asp" />
<add value="index.htm" />
<add value="iisstart.htm" />
<add value="default.aspx" />
</files>
</defaultDocument>
<directoryBrowse enabled="false" />
</system.webServer>

</configuration>
```

This one simply specifies a few things for this site - the list of potential default docs and whether directory browsing is enabled or not.

An important note (and something I just ran into today) is that this file is pretty specific to the server you are on. It's not a great idea to put this file in your source code repository and have it deployed for example. In our case we deployed the production file to staging. The production file had a specific line in it from the production web server that implemented DotNetDefender (a nice URL scoping filter that helps weed out DOS, buffer overruns and other pernicious attacks). Our dev site (which is all internal to our network) doesn't have this filter installed. When the web.config file was deployed it resulted in our requests all returning 404 errors. It took about 20 minutes of head scratching before I figured out what was going on. Imagine how panicked we would be if we had deployed a dev web.config file to production with the same result (yikes!).

Anyway, like many site specific files (ini files, sometimes Application.cfm or .cfc files etc) you should carefully consider whether you want this particular file to become a part of the "official" code base.

Finally, there are many things you can do with the web.config file - much like the venerable httpd.conf file. You can add rules for mod_rewrite, add specific redirects, control permissions for folders etc. It's a very versatile new tool in the IIS arsenal. And

yes Wil, you can manipulate this and all other IIS properties from the command line. Indeed, with Win08r2 Core you don't even need a desktop to be running at all. How does that grab you?