Scripting IIS: Adding a Web Site

Posted At : December 17, 2008 8:30 AM | Posted By : Steve
Related Categories: ColdFusion, Web Servers

As part of a larger project, I need the ability to add a new web site. Part of this process required that I create a web site entry in IIS. This ended up requiring a bit of research, so I thought I would post my results.

The process mostly uses VBScript from Microsoft. I had to find scripts from two or three places on their site and combine them to get the whole thing to work as I wanted.

The big challenge was to make sure that the web site was created, started, and publicly accessible.

Here is the final function:

```coldfusion
<cffunction name="addIISWebSite" access="public" returntype="any" output="false" hint="">
   <cfargument name="destination" type="string" required="yes">
   <cfargument name="SiteName" type="string" required="yes">
   <cfargument name="domains" type="string" required="yes">

   <cfset var AddSiteVBS = "">
   <cfset var AddSiteFile = "#arguments.destination#addsite.vbs">
   <cfset var BatchFile = "#arguments.destination#addsite.bat">
   <cfset var BatchCode = "cscript #arguments.destination#addsite.vbs">
   <cfset var ii = 0>
   <cfset var domain = "">
   <cfset var StartSiteVBS = "">
   <cfset var IP = CreateObject("java", "java.net.InetAddress").getLocalHost().getHostAddress()>

<cfsavecontent variable="AddSiteVBS"><cfoutput>
' Make connections to WMI, to the IIS namespace on MyMachine, and to the Web service.
strComputer = "."
set locatorObj = CreateObject("WbemScripting.SWbemLocator")
set providerObj = locatorObj.ConnectServer(strComputer, "root/MicrosoftIISv2")
set serviceObj = providerObj.Get("IIsWebService='W3SVC'")

' Build binding object, which is a required parameter of the CreateNewSite method.
' Use the SpawnInstance WMI method since we are creating a new instance of an object.
Bindings = Array(#ListLen(arguments.domains)-1#)<cfloop from="1" to="#ListLen(arguments.domains)#" index="ii"
step="1"><cfset domain = ListGetAt(arguments.domains,ii)>
Set Bindings(#ii-1#) = providerObj.get("ServerBinding").SpawnInstance_()
Bindings(#ii-1#).IP = "#IP#"
Bindings(#ii-1#).Port = "80"
Bindings(#ii-1#).Hostname = "#domain#"
</cfloop>
' Create the new Web site using the CreateNewSite method of the IIsWebService object.
Dim strSiteObjPath
strSiteObjPath = serviceObj.CreateNewSite("#arguments.SiteName#", Bindings, "#arguments.destination#")
If Err Then
WScript.Echo "*** Error Creating Site: " & Hex(Err.Number) & ": " & Err.Description & " ***"
WScript.Quit(1)
End If

' strSiteObjPath is in the format of IIsWebServer='W3SVC/1180970907'
' To parse out the absolute path, W3SVC/1180970907, use the SWbemObjectPath WMI object.
Set objPath = CreateObject("WbemScripting.SWbemObjectPath")
objPath.Path = strSiteObjPath
strSitePath = objPath.Keys.Item("")
```

```
' Set some properties on the root virtual directory which was created by CreateNewSite.
Set vdirObj = providerObj.Get("IIsWebVirtualDirSetting='" & strSitePath & "/ROOT'")
vdirObj.AuthFlags = 5 ' AuthNTLM + AuthAnonymous
vdirObj.EnableDefaultDoc = True
vdirObj.DirBrowseFlags = &H4000003E ' date, time, size, extension, longdate
vdirObj.AccessFlags = 513 ' read, script
vdirObj.AppFriendlyName = "#arguments.SiteName#"

' Save the new settings to the metabase
vdirObj.Put_()

' CreateNewSite does not start the server, so start it now.
Set serverObj = providerObj.Get(strSiteObjPath)
serverObj.Start

WScript.Echo "A New site called #arguments.SiteName# was created with the path and unique site identification number of
" & strSitePath
</cfoutput></cfsavecontent>

    <cffile action="WRITE" file="#AddSiteFile#" output="#AddSiteVBS#">
    <cffile action="WRITE" file="#BatchFile#" output="#BatchCode#">

    <cfexecute name="#BatchFile#" timeout="999" />

    <cffile action="DELETE" file="#BatchFile#">
    <cffile action="DELETE" file="#AddSiteFile#">

</cffunction>
```

The "destination" argument is the file path of the site. The "SiteName" argument is the name of the site (as I want it to appear in IIS). The "domains" argument is a comma-delimited list of domain names for the site.

First I create a .bat file to call the .vbs file that I will create.

```
<cfset var AddSiteFile = "#arguments.destination#addsite.vbs">
<cfset var BatchFile = "#arguments.destination#addsite.bat">
<cfset var BatchCode = "cscript #arguments.destination#addsite.vbs">
```

When I started developing this, I originally had my batch code as "cscript addsite.vbs". That worked when I executed the .bat with a double-click, but not from CF (as CF executes it from another location). So, I had to specif the full path to the file.

Next I needed to get the IP address of the server on which the code is executing (for my purposes, that is where I need to create the site). Fortunately, **Todd Rafferty** pointed me to some Java code for this - which is also found in **getServerIP()** by Robert Everland III.

```
<cfset var IP = CreateObject("java", "java.net.InetAddress").getLocalHost().getHostAddress()>
```

As I mentioned earlier, the VBScript itself is a combination of a couple of scripts written by Microsoft and I won't claim to be very good with VBScript. I will point out, however, that the virtual directory is essential in order for the site to be publicly accessible.

After that, it is a simple matter of writing the .vbs and .bat files and using cfexecute to run the .bat file (which will, in turn, run the .vbs code). Then I delete both files.

```
<cffile action="WRITE" file="#AddSiteFile#" output="#AddSiteVBS#">
<cffile action="WRITE" file="#BatchFile#" output="#BatchCode#">
```

```
<cfexecute name="#BatchFile#" timeout="999" />

<cffile action="DELETE" file="#BatchFile#">
<cffile action="DELETE" file="#AddSiteFile#">
```

The upshot of all of this is that I can now programmatically create an IIS web site. I suspect there is a better way, but I have yet to find it (if you know of one, I would love to learn it). In the mean time, the problem is solved.

Next up, Apache...