

RE: Adding a virtual FTP folder to IIS

Source: <http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework/2004-04/0157.html>

From: Peter Huang (*v-phuang_at_online.microsoft.com*)

Date: 04/02/04

Date: Fri, 02 Apr 2004 06:57:08 GMT

Hi,

Based on my understanding, you have enable windows authentication and impersonate = ture in your application's web.config and you still get the access denied error.

I think this may be caused by that the anonymous access is not disabled in the IIS.

I think we can follow the Form Authentication modal. So you can change the <authentication mode="Windows" /> back to <authentication mode="Forms" /> and change the <identity impersonate = "true"/> back to <identity impersonate = "false"/>.

Now I assume your web application will run fine with form authentication.

Now we can change the <processModel> Element in the machine.config file(which is a xml file) to username = System. By default the asp.net application will use the ASPNET account. If we change the username attribute to system, the asp.net appliation will running under the process to have full administrative privileges.

If present, the userName attribute runs the ASP.NET worker process with a Windows identity different from that of the default process identity. By default, userName is set to the special value Machine, and the process runs under a user account named ASPNET that is created automatically when ASP.NET is installed. The password for the ASPNET account is cryptographically generated at the time of installation. If valid credentials are presented in the userName and password attributes, the process is run with the given account. One other special value for userName is System, with the password AutoGenerate, which runs the process as an administrative account and allows all ASP.NET user code running under the process to have full administrative privileges. See the Remarks section below for information about using ASP.NET on a server that is a domain controller.

userName and password are stored in clear text in the configuration file. Although IIS will not transmit .config files in response to a user agent request, configuration files can be read by other means, for instance by an authenticated user with proper credentials on the domain that contains the

microsoft.public.dotnet.framework: RE: Adding a virtual FTP folder to IIS

server. For security reasons, the processModel section supports storage of encrypted userName and password attributes in the registry. The credentials must be in REG_BINARY format encrypted by the Windows 2000 and Windows XP Data Protection API (DPAPI) encryption functions. For more information, see the Remarks and Example sections below.

We can find the machine.config file in the path below usually. (you may need to change the path according to your senario.

C:\WINDOWS\Microsoft.NET\Framework\v1.1.4322\CONFIG

For detailed information about the <processModel> Element please refer to the link below.

<processModel> Element

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpgenref/html/gnrfProcessmodelSection.asp>

If this still does not help you, please try to run the code to see under which user account does the code run.

System.Security.Principal.WindowsIdentity.GetCurrent().Name

e.g. you can run the code in the page_load event and write out the value System.Security.Principal.WindowsIdentity.GetCurrent().Name to monitor it.

306158 INFO: Implementing Impersonation in an ASP.NET Application

<http://support.microsoft.com/?id=306158>

Best regards,

Peter Huang

Microsoft Online Partner Support

Get Secure! – www.microsoft.com/security

This posting is provided "AS IS" with no warranties, and confers no rights.