

Re setting file permissions...

Source:

<http://www.tech-archive.net/Archive/Internet-Server/microsoft.public.inetserver.asp.db/2004-03/0682.html>

From: Fran Tirimo (*fween_at_gmx.co.uk*)

Date: 03/23/04

Date: Tue, 23 Mar 2004 19:14:18 -0000

I am developing a small website using ASP scripts to format data retrieved from an Access database. It will run on a Windows 2003 server supporting FrontPage extensions 2002 hosted by the company 1&1 with only limited server configuration via a web based control panel.

My query relates to the ASP security model and how it relates to FrontPage options for setting file access on a database file. If you know of any online documentation covering the following issues I would be grateful for links to it.

For various reasons (session logging, synchronising data with a master database etc) I need to be able to write data to the database file. Essentially the problem is this: how do I make this possible without giving public access to the database file?

The 1&1 control panel allows me to set Read/Write/Execute/Delete permissions for "IUSR" (I assume to mean anonymous unauthenticated users) and "NETWORK" users (I am not sure what this refers to yet).

So far the only way I have managed to write to the database file from an ASP script is by using the control panel to set the following IUSR permissions on the database file:

Read = true (cannot alter this)
Write = true
Execute = true
Delete = false

I understand that when an ASP script executes, it takes on a "security context" corresponding to the user requesting the page. In my case, when an anonymous user makes a request that needs to be logged to the database for example, the script that performs the logging has anonymous user privileges. Therefore for the write to the database to be successful, IUSR write permissions must be allowed.

Of course, these permission settings have some undesirable side effects:

microsoft.public.inetserver.asp.db: Re setting file permissions...

1 – With "Read = true" on the database file for anonymous users means that anyone can download the database file.

2 – With "Write = true" anonymous users could in theory directly alter the database file eg using telnet or some similar method.

I am not sure if this this second point is true...

– What exactly do anonymous write permissions on a file/directory allow to occur?

– I am not yet sure if the server is set up to ONLY accept HTTP GET and POST commands from anonymous users but have contacted the hosting company about this.

I may have found a solution to the above issues. As I am using FrontPage to publish the site I may be able to take advantage of FP's file permission settings by disallowing browsing by anonymous users for the directory containing the database file.

What I need to know is whether my suggested solution to the problem really prevents anonymous users from downloading/modifying my database file and how would I test this?

I assume that I still have to allow IUSR read/write access to the database file, given the ASP security issues mentioned above. (Are FP permissions considered by the web server instead of or in conjunction with Windows file permissions when deciding if access to a resource is to be allowed?)

To check my understanding of FP folder permissions...

– Allow files to be browsed – does this prevent downloading of files in the folder as well as preventing folder contents from being viewed from ANY HTTP client?

– Allow scripts to be run – does this refer to anonymous users being allowed to view pages generated by scripts within the folder? Or does it refer to scripts in other folders access files/data in the said folder?

– Allow programs to be run – this is probably not relevant to me and I should probably disallow this option.

Also, I have read that you can create a "hidden" folder in FP just by giving it a name beginning with an underscore. Would such a folder allow a script to access/write to a database file? My tests seem to show that the _private directory created by default in new FP webs does NOT allow scripts to write to files.

Finally, is there any way within an ASP script to assume a more privileged security context? For example, would using Server.Execute to call another script containing the code to modify the database use a more privileged security context?

Thanks for your help

microsoft.public.inetserver.asp.db: Re setting file permissions...

Francesco Tirimo
fween@gmx.co.uk