Server Error in OMA Application

When you try to access Outlook Mobile Access on the Exchange 2003 Server at http: *serverIP/oma/, you may receive the following error message: !~~#FF0000:Server Error in '/OMA' Application.~~ ~~#FF0000:Access to the path “C:\WINDOWS\Microsoft.NET\Framework\v1.1.4322\Temporary ASP.NET Files\oma\55aaeb43\5ef66257” is denied. ~~* Description: *An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.* Exception Details: *System.UnauthorizedAccessException: Access to the path “C:\WINDOWS\Microsoft.NET\Framework\v1.1.4322\Temporary ASP.NET Files\oma\55aaeb43\5ef66257” is denied. ASP.NET is not authorized to access the requested resource. Consider granting access rights to the resource to the ASP.NET request identity. ASP.NET has a base process identity (typically {MACHINE}\ASPNET on IIS 5 or Network Service on IIS 6) that is used if the application is not impersonating. If the application is impersonating via <identity impersonate=“true”/>, the identity will be the anonymous user (typically IUSR_MACHINENAME) or the authenticated request user. To grant ASP.NET write access to a file, right-click the file in Explorer, choose “Properties” and select the Security tab. Click “Add” to add the appropriate user or group. Highlight the ASP.NET account, and check the boxes for the desired access.* Source Error: *^An unhandled exception was generated during the execution of the current web request. Information regarding the origin and location of the exception can be identified using the exception stack trace below.^* Stack Trace: *^{UnauthorizedAccessException: Access to the path “C:\WINDOWS\Microsoft.NET\Framework\v1.1.4322\Temporary ASP.NET Files\oma\55aaeb43\5ef66257” is denied.} System.IO.*Error.WinIOError(Int32 errorCode, String str) +393 System.IO.Directory.InternalCreateDirectory(String fullPath, String path) +632 System.IO.Directory.CreateDirectory(String path) +195 System.Web.Compilation.PreservedAssemblyEntry.DoFirstTimeInit(HttpContext context) +85 System.Web.Compilation.PreservedAssemblyEntry.EnsureFirstTimeInit(HttpContext context) +97 System.Web.Compilation.PreservedAssemblyEntry.GetPreservedAssemblyEntry(HttpContext context, String virtualPath, Boolean fApplicationFile) +29 System.Web.UI.TemplateParser.GetParserCacheItemFromPreservedCompilation() +91 System.Web.UI.TemplateParser.GetParserCacheItemInternal(Boolean fCreateIfNotFound) +178 System.Web.UI.TemplateParser.GetParserCacheItemWithNewConfigPath() +125 System.Web.UI.TemplateParser.GetParserCacheItem() +99 System.Web.UI.ApplicationFileParser.GetCompiledApplicationType(String inputFile, HttpContext context, ApplicationFileParser& parser) +171 System.Web.HttpApplicationFactory.CompileApplication(HttpContext context) +43 System.Web.HttpApplicationFactory.Init(HttpContext context) +485 System.Web.HttpApplicationFactory.GetApplicationInstance(HttpContext context) +170 System.Web.HttpRuntime.ProcessRequestInternal(HttpWorkerRequest wr) +414^ ——————————————————————————– *Version Information:* Microsoft .NET Framework Version:1.1.4322.2300; ASP.NET Version:1.1.4322.2300 ——————————————————————————– *To Resolve the problem so that this error message will not appear and OMA displays correctly.* See: [http://support.microsoft.com/kb/818486|http://support.microsoft.com/kb/818486] and reprinted in part below. Article ID: 818486 - Last Review: October 25, 2007 - Revision: 2.4 According to this Microsoft Knowledge Base Article 818486: This problem occurs when you install the Microsoft .NET Framework on a server that is running Microsoft Windows 2000 or when you install Microsoft ASP.NET on a server that is running Microsoft Windows Server 2003 before you promote the server to the role of domain controller. When you promote the server, the access control list (ACL) permissions that you

configured during the .NET Framework or during the ASP.NET installation are modified, and all ASP.NET applications are affected by this change. You must install ASP.NET after you promote the computer to a domain controller for applications that require ASP.NET. This problem may also occur if you upgrade a Windows 2000 domain controller to a Windows Server 2003 domain controller. Note: A configuration where Exchange Server 2003 is installed on a domain controller is sometimes referred to as a onebox configuration. *To resolve this problem, use the ASP.NET IIS Registration Tool (Aspnet_regiis.exe) to reset the ASP.NET ACLs. To run Aspnet_regiis.exe, follow these steps:* 1. On the domain controller, click Start, and then click Run. 2. In the Open box, type cmd, and then click OK. 3. Type cd drive:\Windows\Microsoft.NET\Framework, where drive is the drive that Windows is installed on, and then press ENTER. ^CORRECTION to Support Article: Instead Type *cd\Windows\Microsoft.NET\Framework*, and then press enter.^ 4. Type dir, and then press ENTER. 5. Note the folder names that are listed, and then locate the folder name that begins with the letter v. This folder indicates the version of .NET Framework that is installed on your computer. 6. Type cd VersionFolderName, where VersionFolderName is the folder that you noted in step 5, and then press ENTER. ^CORRECTION to Support Article: Instead Type *cd\Windows\Microsoft.NET\Framework\v1.1.4322* (if that is your .NET version number) and then press ENTER. (Familiarity with MS DOS command line directory navigation is a plus.)^ 7. Type aspnet_regiis.exe -i, and then press ENTER. ————————– To work around this problem, install Exchange Server 2003 on a server that is running Windows 2000 in the following sequence: 1. Install Windows 2000. 2. Promote the server to the role of domain controller. 3. Install the .NET Framework. 4. Install Exchange Server 2003. To work around this problem, install Exchange Server 2003 on Windows Server 2003 in the following sequence: 1. Install Windows Server 2003. 2. Promote the server to the role of domain controller. 3. Install ASP.NET. 4. Install Exchange Server 2003. ———————— After you re-apply ASP.NET to IIS, you will likely receive the following message when you attempt to access and logon to OMA from a web browser. ^~~~#FF0000:Your user account has not been enabled for wireless access. Please contact your system administrator for additional assistance.~~~^ We will look at handling this while configuring exchange. *To fix 'Your user account has not been enabled for wireless access'*, Open Exchange System Manager, expand Global Settings, right-click Mobile Services and select Properties. On the General Tab, under Outlook Mobile Access, Check the box that states “Enable Outlook Mobile Access, and click the Apply button, and click OK button. Now, when you try to reach OMA from Internet Explorer, you receive the following message: ^The device type you are using is not supported. User Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)^ *You may not want to fix this.* Your mobile device may have the ability to access OMA without making any additional changes. However, *to fix this to view the OMA in Internet Explorer (and from certain unsupported devices)__, you merely return to the same Mobile Service Properties, and on the General Tab click the checkbox that states “Enable Unsupported Devices” and click Apply and click OK. Go test it now from IE. Then come back and disable the access from Unsupported Devices like IE.*

From:
https://www.installconfig.com/ - **Install Config Wiki**

Permanent link:
https://www.installconfig.com/doku.php?id=wiki:server_error_in_oma_application&rev=1509938192

Last update: **2017/11/06 03:16**