How to Get Your Site Online

2017-02-14

Orckestra, Europe
Nygårdsvej 16
DK-2100 Copenhagen
Phone +45 3915 7600

www.orckestra.com

Contents

1 Getting Your Website Online

Once you have created a C1 CMS website locally, you normally want to make it available online, that is, publish it.

Basically, all you need is web hosting that allows you to run the ASP.NET version required by C1 CMS (see “System Requirements”) - in Full Trust mode.

Please note that not all web hosting providers offer ASP.NET services or allow running ASP.NET applications in Full Trust mode. You should check these prerequisites with your provider before publishing the website.

Publishing a C1 CMS website is pretty straightforward. You should upload the files of your C1 CMS website to your web hosting provider via FTP or using any other upload method.

Please note that if your C1 CMS website is running on WebMatrix, you may as well use WebMatrix’s publishing feature.

In this quick deployment guide we will consider three ways of publishing your C1 CMS website:

- Publishing using WebMatrix
- Publishing to your web hosting provider via FTP
- Publishing to your own IIS

Should you have any trouble while publishing your website, check the Troubleshooting section at the end of this guide.

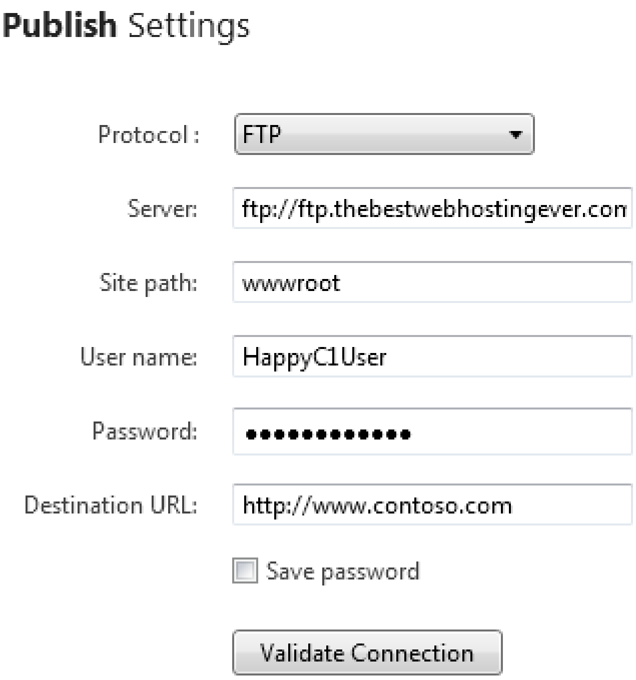
1.1 Publishing Using WebMatrix

Being a good all-in-one platform to try a Web application such as C1 CMS, WebMatrix also allows you to publish websites using its publishing feature.

You will need publishing information from your web hosting provider to configure WebMatrix. Some providers are WebMatrix-aware and can provide you with a Publishing Settings file (.PublishingSetting), which you should import to have WebMatrix configured automatically.

Once configuration is complete, you should select the protocol to use for publishing. WebMatrix offers FTP and Web Deploy.

If you do not have web hosting yet, you can use WebMatrix to find it.



Publish Settings

Protocol : FTP

Server: ftp://ftp.thebestwebhostingever.com

Site path: wwwroot

User name: HappyC1User

Password: ●●●●●●●●●●●●●

Destination URL: http://www.contoso.com

□ Save password

Figure 1: Publishing using WebMatrix

For more information about publishing your website with WebMatrix, please refer to WebMatrix online Help (for example, “How to Publish” and the like).

Please note that even though you run your C1 CMS website on WebMatrix, you may as well publish it manually via FTP as described in “Publishing to Your Web Hosting Provider via FTP”.

1.2 Publishing to Your Web Hosting Provider via FTP

If you own an account with a web hosting provider, you can publish your C1 CMS website via FTP (or any other method that you provider offers for deploying website).

To successfully run a C1 CMS website on the server, you should make sure with your provider that:

- your web hosting supports ASP.NET applications
- and they can run in Full Trust mode.

Uploading a website via FTP is a simple XCOPY operation:

1. Connect to your web hosting provider via FTP with your credentials using your favorite FTP client (e.g. FileZilla).
2. Copy (upload) all the folders and files to the root folder.

Once done, you can access you website online by the URL given to you by your provider.

If you run your local website on WebMatrix, consider “Publishing Using WebMatrix”.

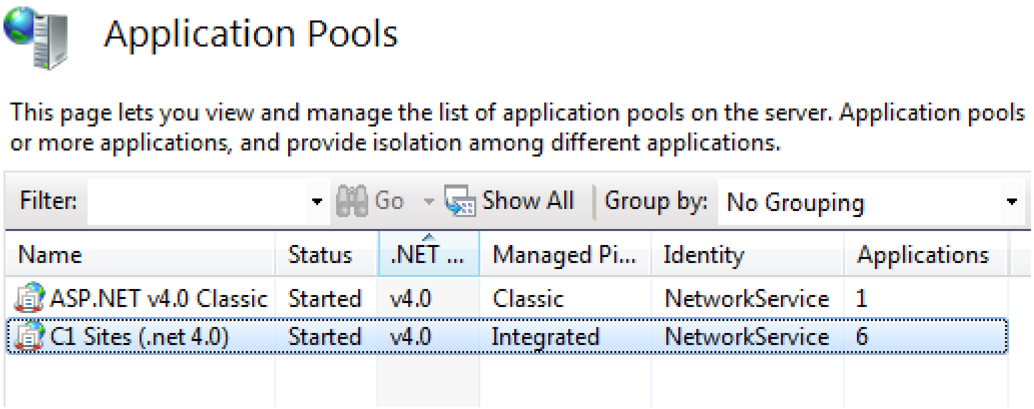
Many providers allow you to upload your website packed as a ZIP archive, and they will unzip it on the server for you. If so, you should zip your website before uploading it. We recommend using our ad-hoc add-on for this purpose: XML-Based Site Backup.

Please note that you should copy the files and folders in the root of your local website to the root folder on the remote server. If you upload your files in a subfolder on the remote server, you would need to take additional steps to configure your remote website subfolder to work as an application provided that your hosting plan allows this.

1.3 Publishing to Your Own IIS

If you have access or own a remote server with IIS, you can publish a local C1 CMS website to this IIS. The following steps are intended for IIS version 7.5 or later.

1. Create or make sure to have an application pool with the .NET Framework version required by C1 CMS (see “System Requirements”).
2. We highly recommend setting its managed pipeline mode to “Integrated” (although C1 CMS may as well run in “Classic” mode).



Application Pools

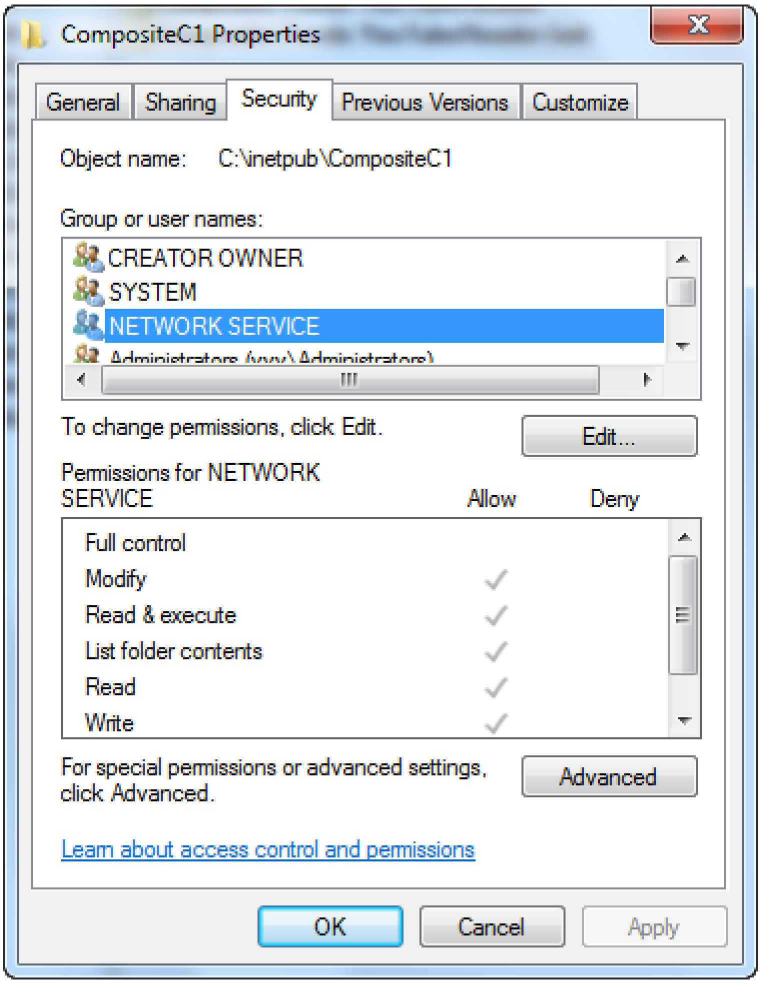
This page lets you view and manage the list of application pools on the server. Application pools or more applications, and provide isolation among different applications.

Group by: No Grouping

Name	Status	.NET ...	Managed Pi...	Identity	Applications
ASP.NET v4.0 Classic	Started	v4.0	Classic	NetworkService	1
C1 Sites (.net 4.0)	Started	v4.0	Integrated	NetworkService	6

Figure 2: Configuring an application pool on IIS 7 for C1 CMS

3. Copy your website files to a folder on the remote server (XCOPY).
4. For the identity the application pool is running under (for example, “NETWORK SERVICE”), on this with your website, set permissions to allow modifying, reading, writing etc.



Object name: C:\inetpub\CompositeC1

Group or user names:

- CREATOR OWNER
- SYSTEM
- NETWORK SERVICE
- Administrators ([illegible]\Administrators)

To change permissions, click Edit.

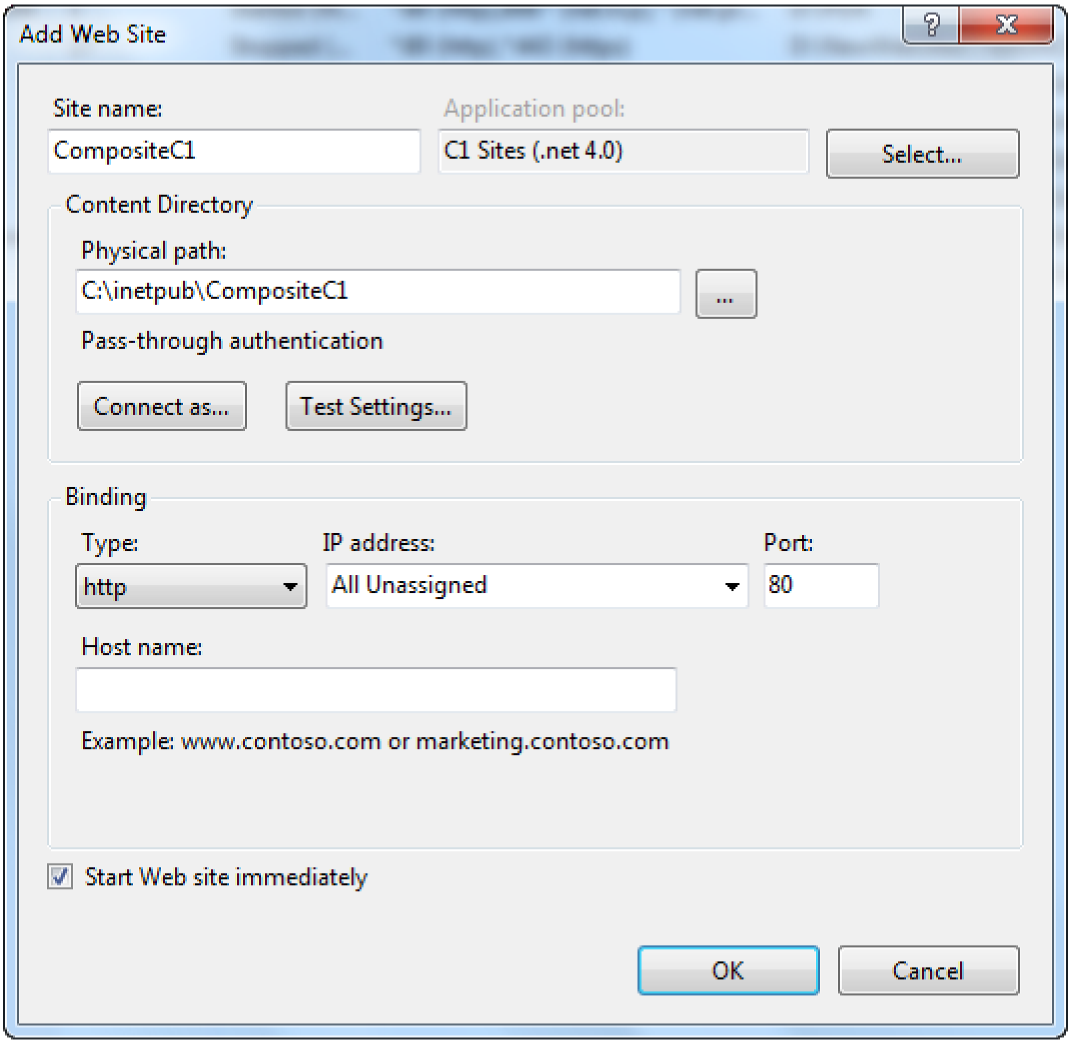
Permissions for NETWORK SERVICE	Allow	Deny
Full control		
Modify	✓	
Read & execute	✓	
List folder contents	✓	
Read	✓	
Write	✓	

For special permissions or advanced settings, click Advanced.

Learn about access control and permissions

Figure 3: Setting read and write permissions on the website directory

5. Create a website on IIS (“Add Web Site...”), selecting the application pool from Step 1 and point it to the website’s folder from Step 3.



Add Web Site

Site name: CompositeC1

Application pool: C1 Sites (.net 4.0)

Content Directory

Physical path: C:\inetpub\CompositeC1

Pass-through authentication

Binding

Type: http

IP address: All Unassigned

Port: 80

Host name:

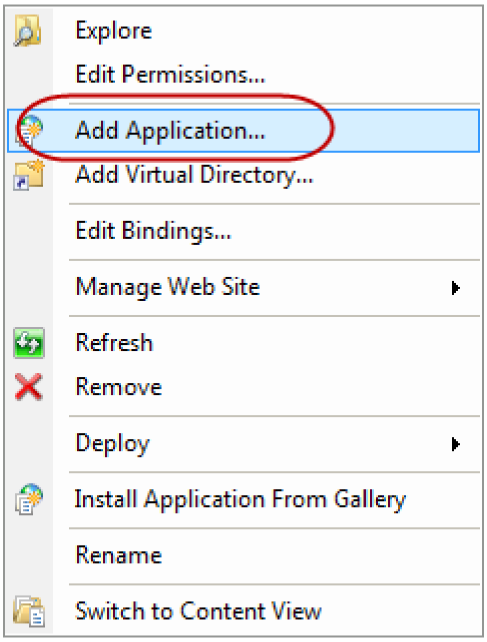
Example: www.contoso.com or marketing.contoso.com

☑ Start Web site immediately

Figure 4: Creating a C1 CMS website on IIS 7

6. If needed, specify a port other than 80 and the host name.
7. Save the configuration (OK).

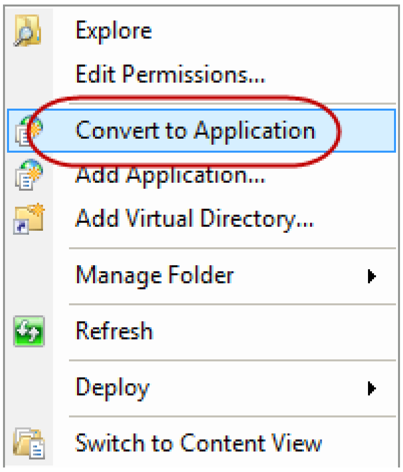
If you run your C1 CMS website in a subfolder on IIS, make sure that it works as an application, and not as a Virtual Directory. You simply need to select “Add Application” in the context menu of the root website you add a C1 CMS website to.



Explore
Edit Permissions...
Add Application...
Add Virtual Directory...
Edit Bindings...
Manage Web Site
Refresh
Remove
Deploy
Install Application From Gallery
Rename
Switch to Content View

Figure 5: Adding a subfolder as an Application

If you have already added a C1 CMS website as a virtual directory, you can still convert it to an application, selecting “Convert to Application” in the context menu.



Explore
Edit Permissions...
Convert to Application
Add Application...
Add Virtual Directory...
Manage Folder
Refresh
Deploy
Switch to Content View

Figure 6: Converting a Virtual Directory into an Application

2 Troubleshooting

Here you can learn about fixing common errors encountered after publishing C1 CMS websites.

To successfully troubleshoot errors, you should first turn on detailed ASP.NET error messages.

2.1 Turning On Detailed ASP.NET Error Messages

A good start to finding a fix to an error is turn on the detailed ASP.NET error messages in web.config and re-publish web.config again.

In most cases, you will access your website remotely. By default, remote users will see custom errors (“RemoteOnly”), which are not informative. You have to turn off custom errors (“Off”). This automatically enables detailed ASP.NET errors.

1. Edit web.config on the local copy of your website.
2. Locate the element: configuration/system.web/customErrors
3. Set the value of its **mode** attribute to “Off”.

```
<configuration>
   <system.web>
       <customErrors mode="Off" />
   </system.web>
</configuration>
```

Listing 1: Turning on detailed error messages in web.config

4. Save the file.
5. Re-publish web.config to your hosting server.
6. Reopen the website.

When you start the website, you will be able to see detailed information about errors if any.

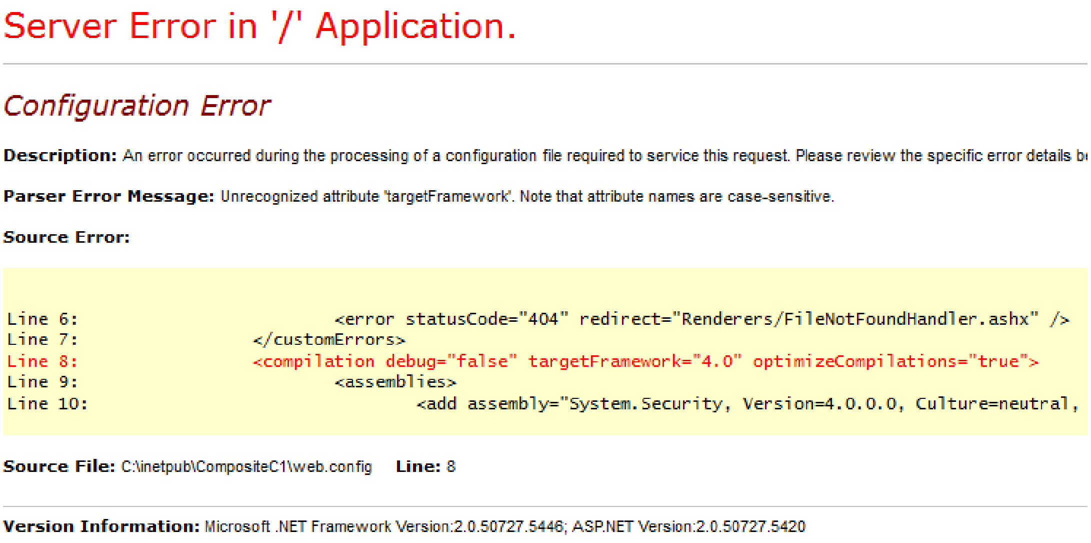
Please note that once the error is detected and fixed, you should turn off the detailed error messages, that is, turn the custom errors back on. For this, repeat Steps 1-5 but this time set the value to “RemoteOnly”.

The following is the list of typical errors you may encounter after publishing your C1 CMS website to your Web hosting.

- Wrong .NET version
- No sufficient permissions on the website directory
- No write permissions on the website directory
- Not run as an application on IIS

2.2 Wrong .NET Version

When you open your website, you can get the server error: “Unrecognized attribute 'targetFramework'”.



Server Error in '/' Application.

Configuration Error

Description: An error occurred during the processing of a configuration file required to service this request. Please review the specific error details b

Parser Error Message: Unrecognized attribute 'targetFramework'. Note that attribute names are case-sensitive.

Source Error:

```
Line 6:                 <error statusCode="404" redirect="Renderers/FileNotFoundHandler.ashx" />
Line 7:             </customErrors>
Line 8:             <compilation debug="false" targetFramework="4.0" optimizeCompilations="true">
Line 9:                 <assemblies>
Line 10:                    <add assembly="System.Security, Version=4.0.0.0, Culture=neutral,
```

Source File: C:\inetpub\CompositeC1\web.config   **Line:** 8

Version Information: Microsoft .NET Framework Version:2.0.50727.5446; ASP.NET Version:2.0.50727.5420

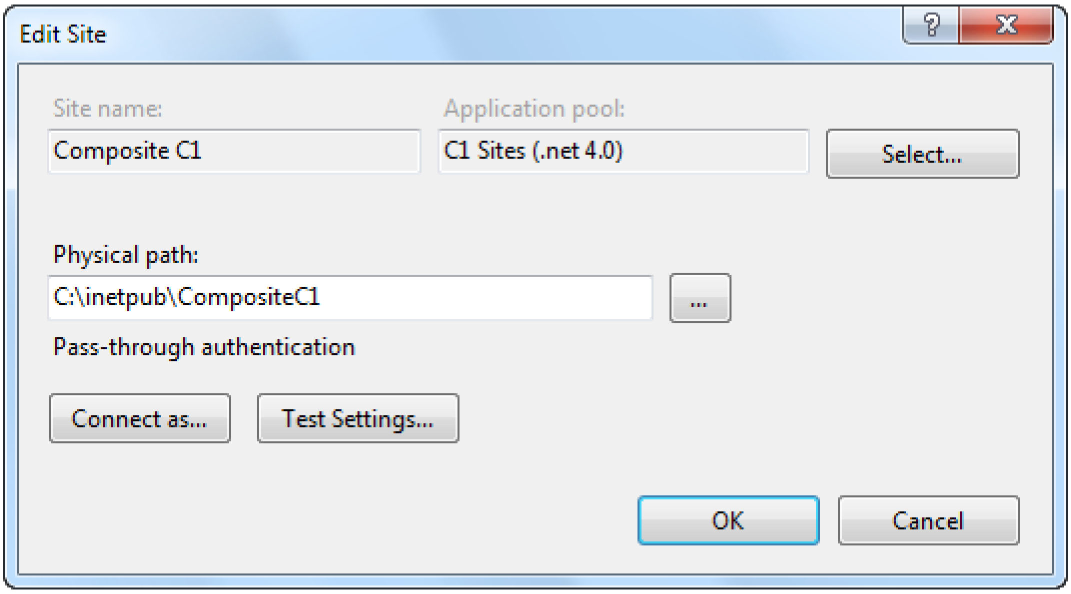
Figure 7: Error when running on the wrong .NET version

This error occurs if your website is running on the wrong .NET version.

If you have access to IIS, do as described below. Otherwise, follow the standard procedure of changing the .NET Platform version on your web hosting or contact your web hosting provider.

On IIS 7:

1. Edits your website Basic Settings.
2. In the Application Pool field, select the application pool with the .NET Framework version required by C1 CMS (see "System Requirements").



Edit Site

Site name: Composite C1

Application pool: C1 Sites (.net 4.0)

Physical path: C:\inetpub\CompositeC1

Pass-through authentication

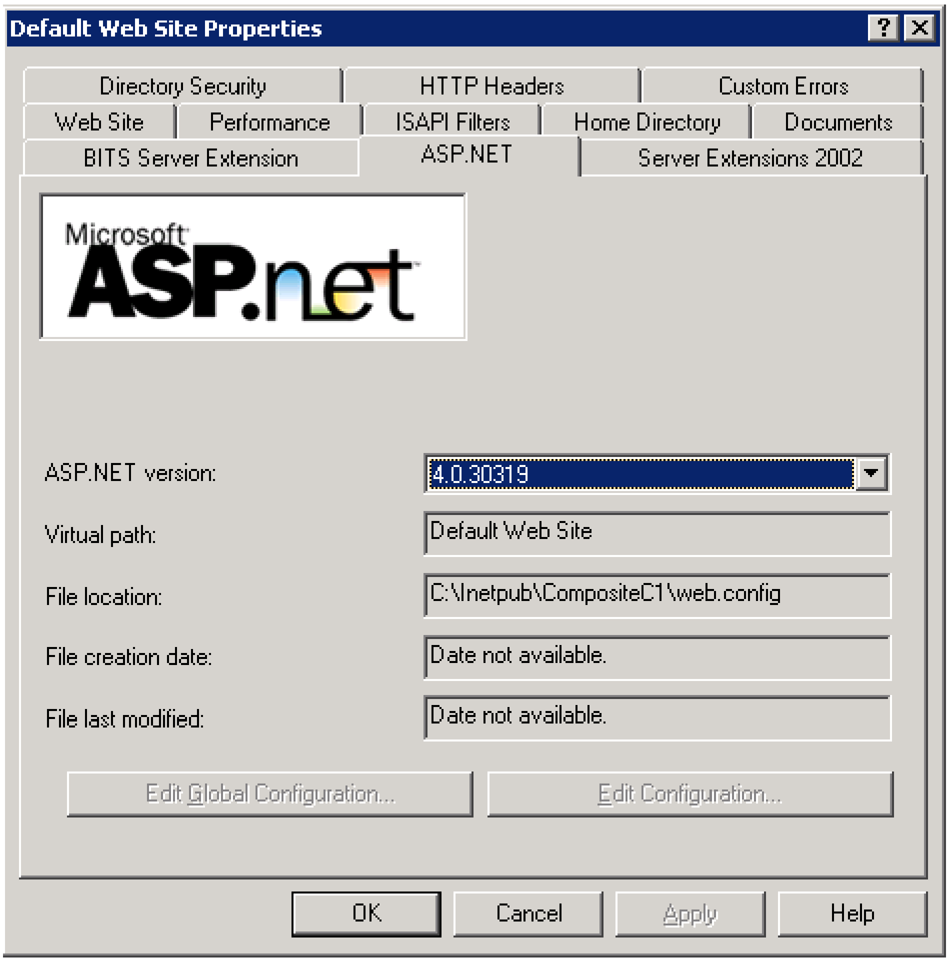
Figure 8: Selecting an application pool with .NET Framework on IIS 7

3. If you do not have such a pool, create it.

On IIS 6:

1. Open the Properties of your website.
2. Open the ASP.NET tab.

3. In the ASP.NET version field, select “4.0.319” (or a similar value - please see “System Requirements”)



ASP.NET version: 4.0.30319

Virtual path: Default Web Site

File location: C:\Inetpub\CompositeC1\web.config

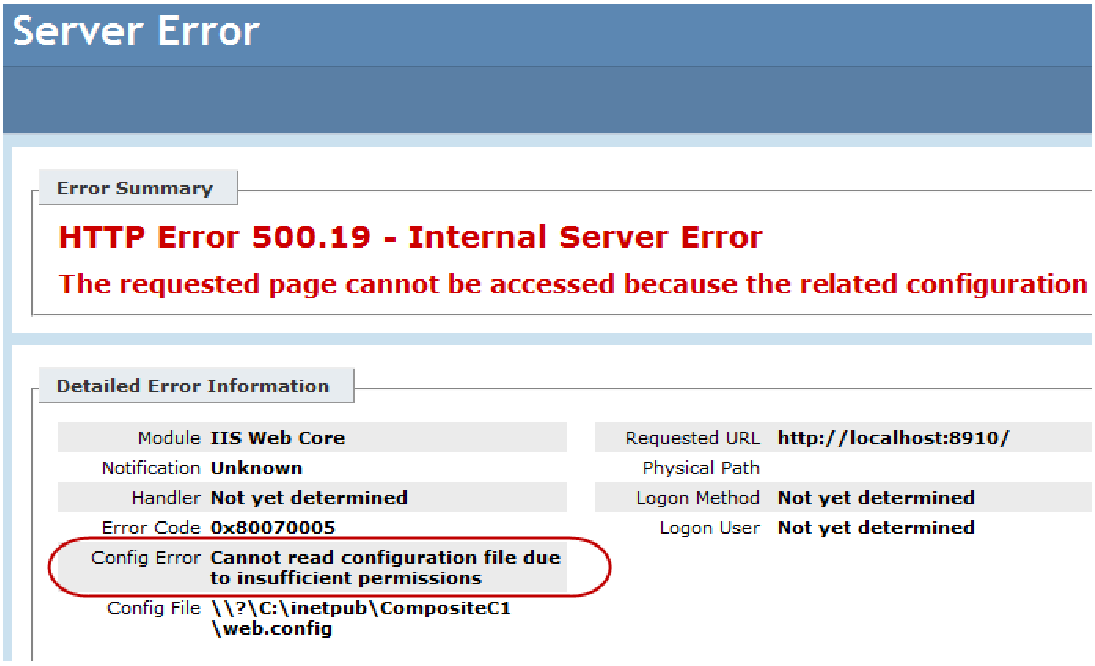
File creation date: Date not available.

File last modified: Date not available.

Figure 9: Selecting the ASP.NET version for a C1 CMS website on IIS 6

2.3 No Sufficient Permissions on the Website Directory

When you open your website, you can get the server error: “HTTP 500.19”, “Cannot read configuration file due to insufficient permissions”.



Server Error

Error Summary

HTTP Error 500.19 - Internal Server Error

The requested page cannot be accessed because the related configuration

Detailed Error Information

Module	IIS Web Core	Requested URL	http://localhost:8910/
Notification	Unknown	Physical Path	
Handler	Not yet determined	Logon Method	Not yet determined
Error Code	0x80070005	Logon User	Not yet determined
Config Error	Cannot read configuration file due to insufficient permissions		
Config File	\\?\C:\inetpub\CompositeC1 \web.config		

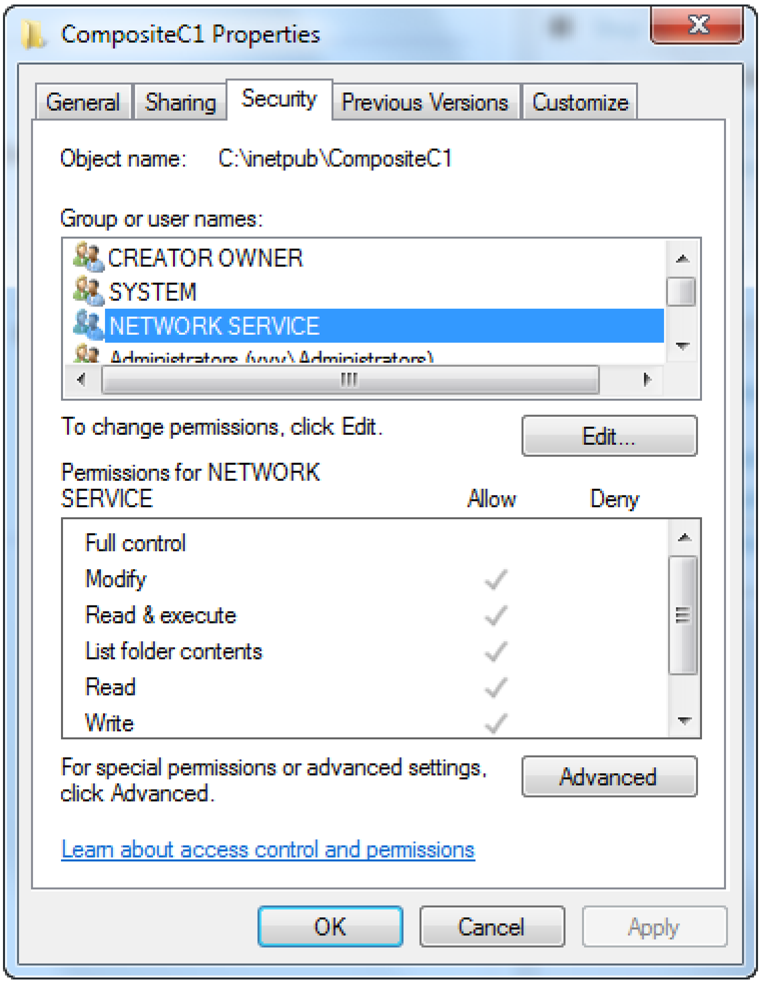
Figure 10: Error when having no sufficient permissions on the website directory

This error occurs if you do not have sufficient (*modify, read, write* etc) permissions on the website directory.

If you have access to IIS, do as described below. Otherwise, follow the standard procedure of setting permissions on website directories on your web hosting or contact your web hosting provider.

On IIS or in Windows explorer:

1. Edit permissions on the website directory.
2. Make sure that **modify**, **execute**, **list**, **read**, and **write** permissions are set for the identity your website is running under. (On IIS 6, it should be “NETWORK SERVICE” under which ASP.NET is running. On IIS 7, it should be the identity under which the application pool used by your site is running.)



Object name: C:\inetpub\CompositeC1

Group or user names:

- CREATOR OWNER
- SYSTEM
- NETWORK SERVICE
- Administrators (www\Administrators)

To change permissions, click Edit.

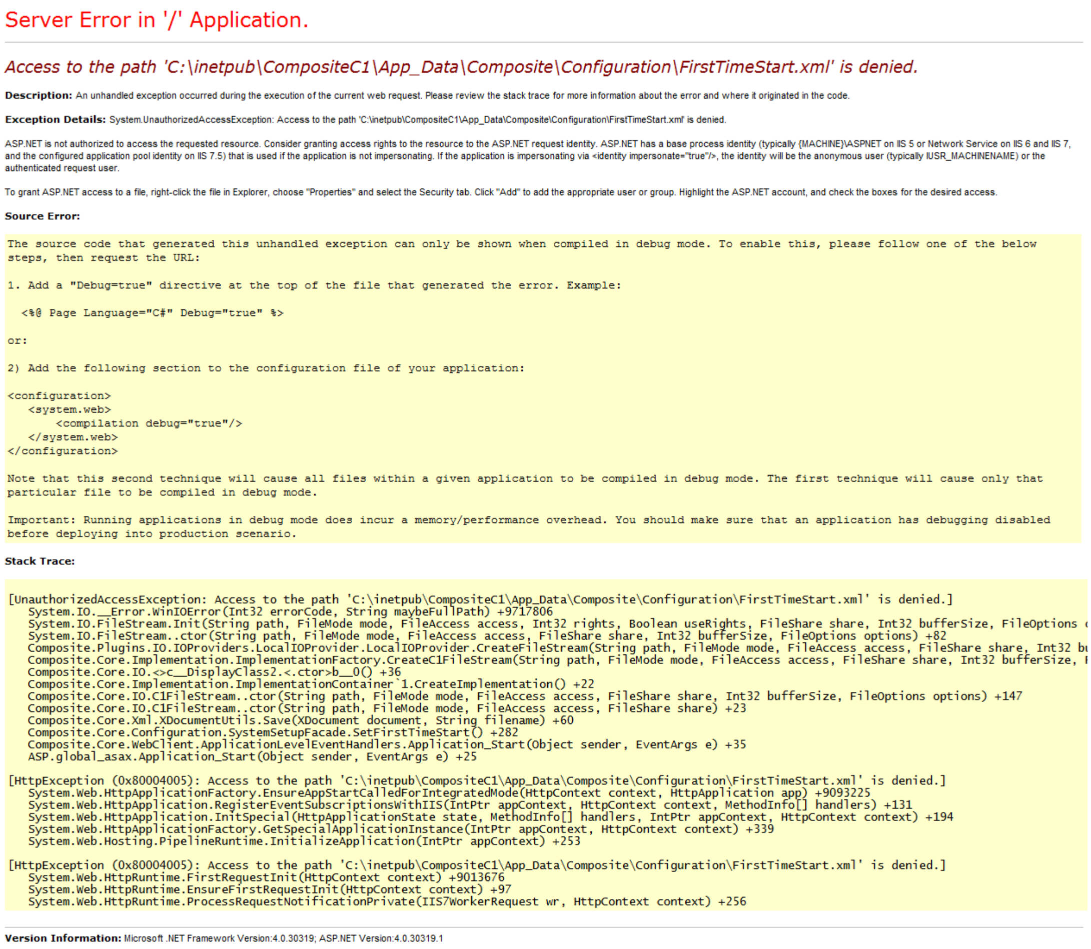
Permissions for NETWORK SERVICE	Allow	Deny
Full control		
Modify	✓	
Read & execute	✓	
List folder contents	✓	
Read	✓	
Write	✓	

For special permissions or advanced settings, click Advanced.

Figure 11: Setting proper permissions on the website directory

2.4 No Write Permissions on the Website Directory

When you open your website, you can get the server error: “Access to the path 'C:\inetpub\C1-CMS\App_Data\Composite\Configuration\FirstTimeStart.xml' is denied.”, or something similar.



Server Error in '/' Application.

Access to the path 'C:\inetpub\CompositeC1\App_Data\Composite\Configuration\FirstTimeStart.xml' 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:\inetpub\CompositeC1\App_Data\Composite\Configuration\FirstTimeStart.xml' 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 and IIS 7, and the configured application pool identity on IIS 7.5) 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 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:

```
The source code that generated this unhandled exception can only be shown when compiled in debug mode. To enable this, please follow one of the below
steps, then request the URL:

1. Add a "Debug=true" directive at the top of the file that generated the error. Example:

  <%@ Page Language="C#" Debug="true" %>

or:

2) Add the following section to the configuration file of your application:

<configuration>
   <system.web>
       <compilation debug="true"/>
   </system.web>
</configuration>

Note that this second technique will cause all files within a given application to be compiled in debug mode. The first technique will cause only that
particular file to be compiled in debug mode.

Important: Running applications in debug mode does incur a memory/performance overhead. You should make sure that an application has debugging disabled
before deploying into production scenario.
```

Stack Trace:

```
[UnauthorizedAccessException: Access to the path 'C:\inetpub\CompositeC1\App_Data\Composite\Configuration\FirstTimeStart.xml' is denied.]
   System.IO.__Error.WinIOError(Int32 errorCode, String maybeFullPath) +9717806
   System.IO.FileStream.Init(String path, FileMode mode, FileAccess access, Int32 rights, Boolean useRights, FileShare share, Int32 bufferSize, FileOptions o
   System.IO.FileStream..ctor(String path, FileMode mode, FileAccess access, FileShare share, Int32 bufferSize, FileOptions options) +82
   Composite.Plugins.IO.IOProviders.LocalIOProvider.LocalIOProvider.CreateFileStream(String path, FileMode mode, FileAccess access, FileShare share, Int32 bu
   Composite.Core.Implementation.ImplementationFactory.CreateC1FileStream(String path, FileMode mode, FileAccess access, FileShare share, Int32 bufferSize, F
   Composite.Core.IO.<>c__DisplayClass2.<.ctor>b__0() +36
   Composite.Core.Implementation.ImplementationContainer`1.CreateImplementation() +22
   Composite.Core.IO.C1FileStream..ctor(String path, FileMode mode, FileAccess access, FileShare share, Int32 bufferSize, FileOptions options) +147
   Composite.Core.IO.C1FileStream..ctor(String path, FileMode mode, FileAccess access, FileShare share) +23
   Composite.Core.Xml.XDocumentUtils.Save(XDocument document, String filename) +60
   Composite.Core.Configuration.SystemSetupFacade.SetFirstTimeStart() +282
   Composite.Core.WebClient.ApplicationLevelEventHandlers.Application_Start(Object sender, EventArgs e) +35
   ASP.global_asax.Application_Start(Object sender, EventArgs e) +25

[HttpException (0x80004005): Access to the path 'C:\inetpub\CompositeC1\App_Data\Composite\Configuration\FirstTimeStart.xml' is denied.]
   System.Web.HttpApplicationFactory.EnsureAppStartCalledForIntegratedMode(HttpContext context, HttpApplication app) +9093225
   System.Web.HttpApplication.RegisterEventSubscriptionsWithIIS(IntPtr appContext, HttpContext context, MethodInfo[] handlers) +131
   System.Web.HttpApplication.InitSpecial(HttpApplicationState state, MethodInfo[] handlers, IntPtr appContext, HttpContext context) +194
   System.Web.HttpApplicationFactory.GetSpecialApplicationInstance(IntPtr appContext, HttpContext context) +339
   System.Web.Hosting.PipelineRuntime.InitializeApplication(IntPtr appContext) +253

[HttpException (0x80004005): Access to the path 'C:\inetpub\CompositeC1\App_Data\Composite\Configuration\FirstTimeStart.xml' is denied.]
   System.Web.HttpRuntime.FirstRequestInit(HttpContext context) +9013676
   System.Web.HttpRuntime.EnsureFirstRequestInit(HttpContext context) +97
   System.Web.HttpRuntime.ProcessRequestNotificationPrivate(IIS7WorkerRequest wr, HttpContext context) +256
```

Version Information: Microsoft .NET Framework Version:4.0.30319; ASP.NET Version:4.0.30319.1

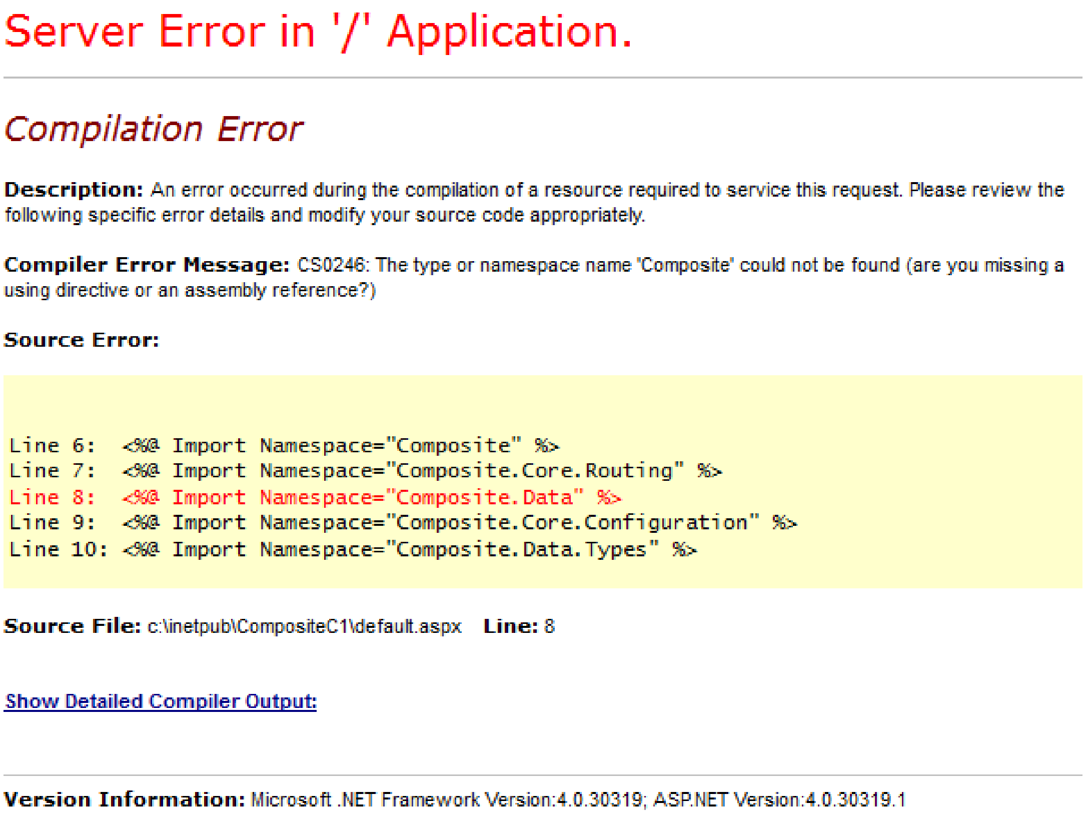
Figure 12: Error when having no write permissions on the website directory

This error occurs if you do not have *write* permissions on the website directory.

To troubleshoot it, do as suggested in “No sufficient permissions on the website directory”.

2.5 Not Run as an Application on IIS

When you open your website, you can get the server errors for example: “CS0246: The type or namespace name 'Composite' could not be found (are you missing a using directive or an assembly reference?)”.



Server Error in '/' Application.

Compilation Error

Description: An error occurred during the compilation of a resource required to service this request. Please review the following specific error details and modify your source code appropriately.

Compiler Error Message: CS0246: The type or namespace name 'Composite' could not be found (are you missing a using directive or an assembly reference?)

Source Error:

```
Line 6:   <%@ Import Namespace="Composite" %>
Line 7:   <%@ Import Namespace="Composite.Core.Routing" %>
Line 8:   <%@ Import Namespace="Composite.Data" %>
Line 9:   <%@ Import Namespace="Composite.Core.Configuration" %>
Line 10:  <%@ Import Namespace="Composite.Data.Types" %>
```

Source File: c:\inetpub\CompositeC1\default.aspx   **Line:** 8

Show Detailed Compiler Output:

Version Information: Microsoft .NET Framework Version:4.0.30319; ASP.NET Version:4.0.30319.1

Figure 13: Error when not running a C1 CMS website as an Application

This error occurs if run your C1 CMS website on IIS 7 as a Virtual Directory. This works on IIS 6, but on IIS 7 you should run a C1 CMS website in a subfolder as an Application.

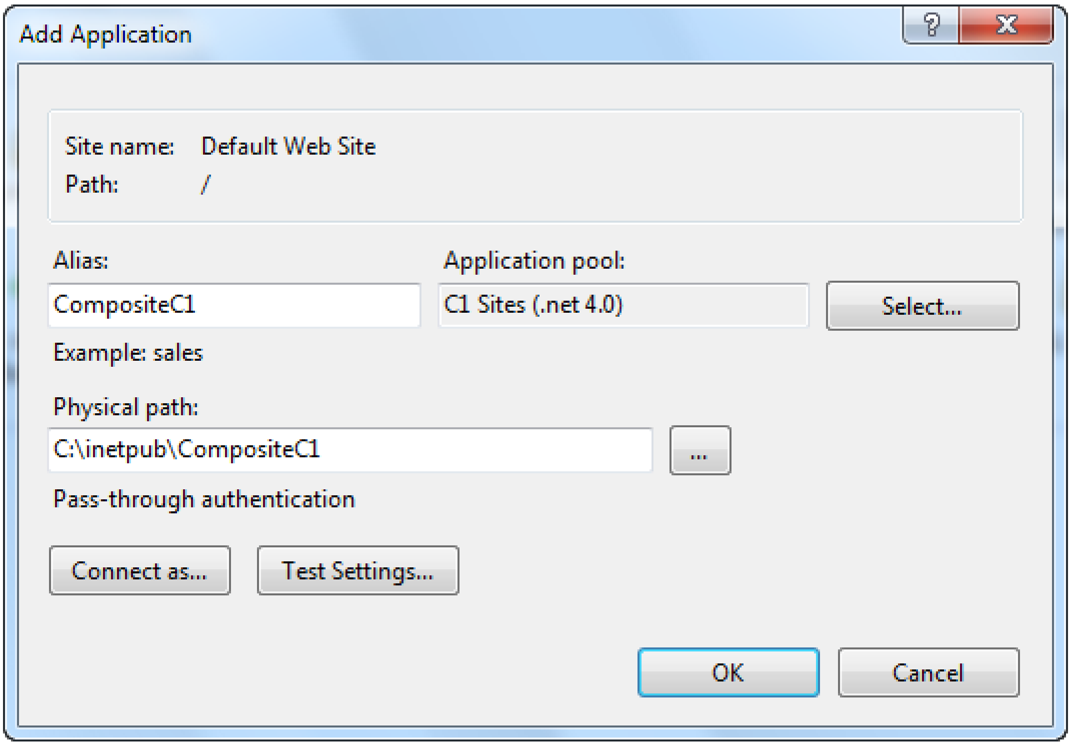
If you have access to IIS, do as described below. Otherwise, make sure that a subfolder you run a C1 CMS website in is configured as an IIS *Application* and not a Virtual Directory by following the standard procedure of configuring your remote website on your web hosting or contact your web hosting provider.

On IIS 7, do one of the following:

1. Right-click the C1 CMS website's subfolder.
2. In the context menu, click Convert to Application.
3. Fill out the fields and click OK.

or

1. Right-click the C1 CMS website you want to create a subfolder in.
2. In the context menu, click Add Application.
3. Fill out the fields and click OK.


Add Application

Site name: Default Web Site
Path: /

Alias: CompositeC1
Example: sales

Application pool: C1 Sites (.net 4.0)

Physical path: C:\inetpub\CompositeC1

Pass-through authentication

Figure 14: Adding a C1 CMS website as an Application in a subfolder