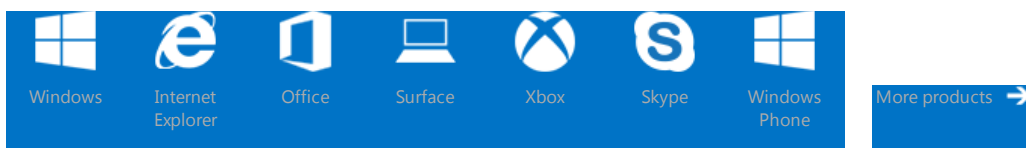


Find it myself ▸

Select the product you need help with

Ask the community

Get live help



# PRB: "Requested Registry Access Is Not Allowed" Error Message When ASP.NET Application Tries to Write New EventSource in the EventLog

Article ID: 329291 - [View products that this article applies to.](#)

This article was previously published under Q329291

## SYMPTOMS

When you use ASP.NET to create a new event source in the event log, you may receive the following error message:

System.Security.SecurityException: Requested registry access is not allowed.

## CAUSE

By default, the user token of the ASP.NET worker process is ASPNET (or NetworkService for applications that run on Internet Information Services [IIS] 6.0). The problem in the "Symptoms" section occurs because your account does not have the correct user rights to create an event source.

## RESOLUTION

**Important** This section, method, or task contains steps that tell you how to modify the registry. However, serious problems might occur if you modify the registry incorrectly. Therefore, make sure that you follow these steps carefully. For added protection, back up the registry before you modify it. Then, you can restore the registry if a problem occurs. For more information about how to back up and restore the registry, click the following article number to view the article in the Microsoft Knowledge Base:

[322756](http://support.microsoft.com/kb/322756/) (<http://support.microsoft.com/kb/322756/>) How to back up and restore the registry in Windows

To resolve this problem, a user who has administrative rights must create the event source before you run the ASP.NET Web Application. To create an event source, use one of the following approaches.

### First Approach

Create an event source under the **Application** event log in Registry Editor. To do this, follow these steps:

1. Click **Start**, and then click **Run**.
2. In the **Open** text box, type **regedit**.
3. Locate the following registry subkey:  
HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\Eventlog\Application
4. Right-click the **Application** subkey, point to **New**, and then click **Key**.
5. Type **TEST** for the key name.
6. Close Registry Editor.

### Second Approach

The **EventLogInstaller** class in the System.Diagnostics namespace permits you to install and configure an event log that your application reads from or writes to while running. You can create an event source by using EventLogInstaller. To do this, follow these steps:

1. Use Microsoft Visual Basic .NET or Microsoft Visual C# .NET to create a new **Class Library** named EventLogSourceInstaller. By default, the **Class1.vb** file or the **Class1.cs** file is created.
2. In Solution Explorer, right-click **EventLogSourceInstaller**, and then click **Add References**.
3. In the **Add Reference** dialog box, double-click **System.Configuration.Install.dll**, and then click **OK**.
4. Rename the Class1.vb\Class1.cs to MyEventLogInstaller.vb\MyEventLogInstaller.cs.
5. Replace the existing code in MyEventLogInstaller.vb or MyEventLogInstaller.cs with the following sample code:

#### Visual Basic .NET Sample

```
Imports System.Diagnostics Imports System.Configuration.Install Imports System.ComponentModel <RunInstaller(True)> _
Public Class MyEventLogInstaller Inherits Installer Private myEventLogInstaller As EventLogInstaller Public Sub New() '
Create an instance of 'EventLogInstaller'. myEventLogInstaller = New EventLogInstaller() ' Set the 'Source' of the event
log, to be created. myEventLogInstaller.Source = "TEST" ' Set the 'Log' that the source is created in.
myEventLogInstaller.Log = "Application" ' Add myEventLogInstaller to 'InstallerCollection'.
Installers.Add(myEventLogInstaller) End Sub End Class
```

#### Visual C# .NET Sample

```
using System; using System.Diagnostics; using System.ComponentModel; using System.Configuration.Install; namespace
EventLogSourceInstaller { [RunInstaller(true)] public class MyEventLogInstaller : Installer { private EventLogInstaller
myEventLogInstaller; public MyEventLogInstaller() { //Create Instance of EventLogInstaller myEventLogInstaller = new
```

```
EventLogInstaller(); // Set the Source of Event Log, to be created. myEventLogInstaller.Source = "TEST"; // Set the Log
that source is created in myEventLogInstaller.Log = "Application"; // Add myEventLogInstaller to the Installers
Collection. Installers.Add(myEventLogInstaller); } }
```

- On the **Build** menu, click **Build Solution** to create **EventLogSourceInstaller.dll**.
- Open the Visual Studio .NET Command Prompt.
- At the command prompt, change to the folder where **EventLogSourceInstaller.dll** is located.
- Run the following command to create the EventSource:  
**InstallUtil EventLogSourceInstaller.dll**

## MORE INFORMATION

### Steps to Reproduce the Behavior

- Use Visual Basic .NET or Visual C# .NET to create a new **ASP.NET Web Application**. By default, **WebForm1.aspx** file is created.
- In the HTML view of **WebForm1.aspx**, replace the existing code with the following sample code:

#### Visual Basic .NET Sample

```
<%@ Page Language="vb" AutoEventWireup="true" %> <%@ Import namespace="System.Diagnostics" %> <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN"> <HTML> <script language="VB" runat="server"> Sub WriteEvent_Click(Src As Object, e As EventArgs) Dim ev As New EventLog("Application") ' Event's Source name ev.Source = "TEST"
EventLog.CreateEventSource(ev.Source, "Application") Try ev.WriteEntry(TextBox1.Text) Catch b as exception Response.write ("WriteEntry " & b.message & "<br>") End Try ev = Nothing End Sub </script> <body> <form id="Form1" runat="server"> Event
message: <asp:textbox id="TextBox1" runat="server" Width="233px"></asp:textbox> <asp:button id="Button1"
onclick="WriteEvent_Click" runat="server" NAME="Button1" text="Write to event log"></asp:button> </form> </body> </HTML>
```

#### Visual C# .NET Sample

```
<%@ Page Language="c#" AutoEventWireup="true" %> <%@ Import namespace="System.Diagnostics" %> <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN"> <HTML> <script language="C#" runat="server"> void WriteEvent_Click(Object Src, EventArgs e) { EventLog ev = new EventLog("Application"); // Event's Source name ev.Source = "TEST";
EventLog.CreateEventSource(ev.Source, "Application"); try { ev.WriteEntry(TextBox1.Text); } catch (Exception b) {
Response.Write("WriteEntry " + b.Message + "<br>"); } ev = null; } </script> <body> <form id="Form1" runat="server"> Event
message: <asp:textbox id="TextBox1" runat="server" Width="233px"></asp:textbox> <asp:button id="Button1"
onclick="WriteEvent_Click" runat="server" NAME="Button1" text="Write to event log"></asp:button> </form> </body> </HTML>
```

- On the **Debug** menu, click **Start** to view the **WebForm1.aspx** page in the browser.
- Type some text in **TextBox**, and then click **Write to event log**.
- The error message that is discussed in the "Symptoms" section of this article appears.
- To resolve this problem, create an Event Source as discussed in the "Resolution" section, and comment the following code in **WebForm1.aspx** :  
  
EventLog.CreateEventSource(ev.Source, "Application")

- Repeat steps 3 and 4.

## REFERENCES

For more information, visit the following Microsoft Web sites:

[http://msdn2.microsoft.com/en-us/library/f5dcf6h3\(vs.71\).aspx](http://msdn2.microsoft.com/en-us/library/f5dcf6h3(vs.71).aspx) ([http://msdn2.microsoft.com/en-us/library/f5dcf6h3\(vs.71\).aspx](http://msdn2.microsoft.com/en-us/library/f5dcf6h3(vs.71).aspx))

[http://msdn2.microsoft.com/en-us/library/system.diagnostics.eventlog\(vs.71\).aspx](http://msdn2.microsoft.com/en-us/library/system.diagnostics.eventlog(vs.71).aspx) ([http://msdn2.microsoft.com/en-us/library/system.diagnostics.eventlog\(vs.71\).aspx](http://msdn2.microsoft.com/en-us/library/system.diagnostics.eventlog(vs.71).aspx))

## Properties

Article ID: 329291 - Last Review: December 3, 2007 - Revision: 2.10

#### APPLIES TO

- Microsoft ASP.NET 1.1
- Microsoft Visual Basic .NET 2003 Standard Edition
- Microsoft Visual C# .NET 2003 Standard Edition
- Microsoft ASP.NET 1.0
- Microsoft Visual Basic .NET 2002 Standard Edition
- Microsoft Visual C# .NET 2002 Standard Edition
- Microsoft Internet Information Services 5.0
- Microsoft Internet Information Services version 5.1
- Microsoft Internet Information Services 6.0

Keywords: kberrmsg kbwebforms kbsecurity kbprb KB329291

## Give Feedback

[↑ Back to the top](#)