Monday, March 24. 2014

Setting up IIS as a reverse proxy

Overview tl;dr; This post is all about setting up IIS(8.5) in order to be used as a reverse proxy. Use multiple different Webservers (IIS/Apache/Tomcat) and run them on the same server and port. Installing IIS Just install IIS by using the add roles and feature functionality of windows server 2012 Installing apache I currently have the problem that I own a windows 2012 virtual server, and need to run IIS and Apache on it. Installation is actually super easy, just install IIS and then Apache (I took the apache binaries for windows from apachehaus) But if you now try to start apache, the following error will show up: (OS 10013)An attempt was made to access a socket in a way forbidden by its access permissions.Â : AH00072: make_sock: could not bind to address [::]:80   This happens because I'm trying to open port 80 with apache while port 80 is already used by IIS.Â And the same would happen when I try to start Tomcat or any other server which tries to run on port 80. This error can be avoided by setting the Listen PortÂ of apache to anything different than port 80, i.e. 8080. Problem with this solution is that any site served by apache can only be reached via port 8080 which is super uncool since a user then has to enter the port 8080 together with the domain name. Setup IIS In order to use IIS as a reverse proxy, the rewrite module must be installed. This can be done by downloading it from the microsoft website To get rewrite(Not only redirect) working, the application request routing must be installed as well from the microsoft website Now the Application Request Routing must be configured in IIS manager. It's as simple as checking the Enable proxy checkbox.   Now a new website must be added in IIS pointing to an empty directory. In order for IIS to act properly as a proxy, this website must be bound to a host name. (For this demo and since I don't have a hostname, I just added a entry into the host file of the computer) When now connecting to this site, noting must be shown since there is no content in the configured folder. Next step is to setup the rewrite rule.   Setup rewrite rule in IIS Add a new Blank rule via URL rewrite   The rule must look like the following sample. This is the corresponding web.config to the screenshot below:

Done
Now it's done. Requests sent to proxytest.com:80 will be rewritten from IIS to the address from rewrite URL.
With this configuration, it's possible to run IIS, Apache, Tomcat and many other webserver at the same time on the same server under the same port.
Great thing.

Posted by suntsu at 19:00

Be careful using ThreadStatic in Visual Studio unit tests

Given is a ThreadStatic field which is set to empty in the static constructor, and set by a public instance method:
```
    public class TestClass
    {
       [ThreadStatic]
       private static string threadStaticValue;
       static TestClass()
       {
          threadStaticValue = "";
```

```
    }
    public void SetThreadStaticValue(string value)
    {
        threadStaticValue = value;
    }
    public string GetThreadStaticValue()
    {
        return threadStaticValue;
    }
}
```

Now two unit tests check whether the threadStaticValue is set correct.

```
[TestClass]
public class ThreadStaticTest
{
    [TestMethod]
    public void WithThreadStaticValueSetTest()
    {
        Debug.WriteLine("Thread id: {0}", System.Threading.Thread.CurrentThread.ManagedThreadId);
        TestClass tc = new TestClass();
        tc.SetThreadStaticValue("foo");
        Assert.AreSame(tc.GetThreadStaticValue(), "foo");
    }
    [TestMethod]
    public void WithThreadStaticValueNotSetTest()
    {
        Debug.WriteLine("Thread id: {0}", System.Threading.Thread.CurrentThread.ManagedThreadId);
        TestClass tc = new TestClass();
        Assert.AreSame(tc.GetThreadStaticValue(), "");
    }
}
```

Those tests will fail(good) or pass(bad) depending of whether the test engine will reuse the same thread.
Executed a second time, the result could be different.
Solution:

Don't use ThreadStatic at all
Make sure the ThreadStatic fields are initialized correctly before using them

```
 Posted by suntsu in .net at 11:28
```

Wednesday, March 12. 2014

Daydeal competition

Daydeal holds a competition where you can win the simpons lego house.   This is my contribution (My first gif).   Cross your fingers! Update Unfortunately, I did not win the first price(simpsons house), but I got a voucher for daydeal nevertheless

 Posted by suntsu in web stuff at 12:20