**Windows Authentication for MVC**

1. Note: Using IIS Express, you are always logged in.
2. Web.config – For Windows Auth

<system.web>

<compilation targetFramework="4.6.1" />

<httpRuntime targetFramework="4.6.1" />

<!-- Windows Auth – Comment this out for no auth -->

<authentication mode="Windows" />

<authorization>

<deny users="?" />

</authorization>

<!-- End Windows Auth -->

</system.web>

Or for a sub folder:

<system.web>

<compilation debug="true" targetFramework="4.6.1" />

<httpRuntime targetFramework="4.6.1" />

<!--<authentication mode="Windows" />

<authorization>

<deny users="?" />

</authorization>-->

<authentication mode="Windows"/>

<authorization>

<allow users="\*"/>

</authorization>

<customErrors mode="On" defaultRedirect="/Home" redirectMode="ResponseRewrite">

<error statusCode="401" redirect="/Home" />

<error statusCode="403" redirect="/Home" />

<error statusCode="404" redirect="/Home" />

<error statusCode="500" redirect="/Home" />

</customErrors>

</system.web>

<location path="Areas/Secure" allowOverride="true">

<system.web>

<authorization>

<deny users="?"/>

</authorization>

</system.web>

</location>

1. Plus for IIS Express, set these two lines in the csproj.

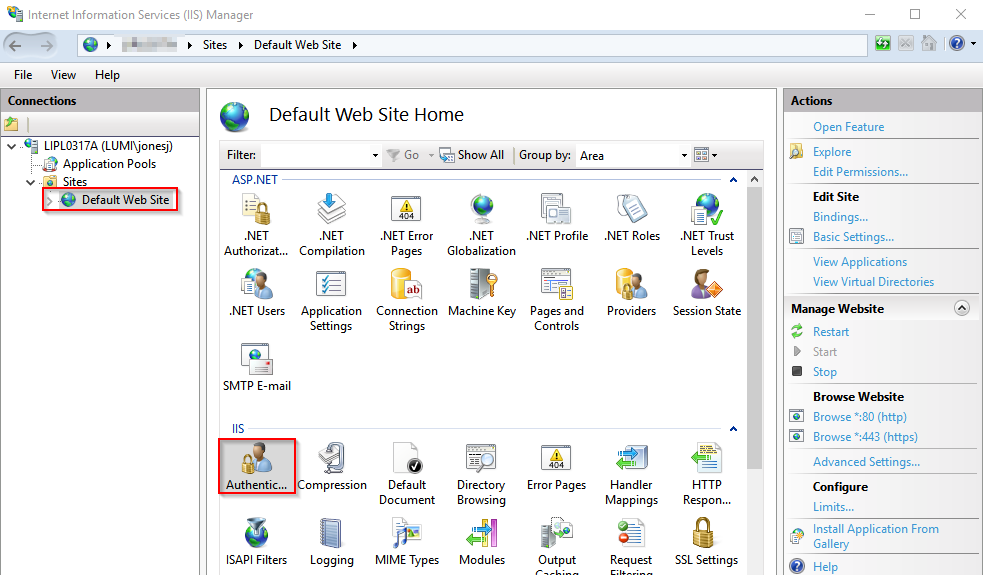
<IISExpressAnonymousAuthentication>disabled</IISExpressAnonymousAuthentication>

<IISExpressWindowsAuthentication>enabled</IISExpressWindowsAuthentication>

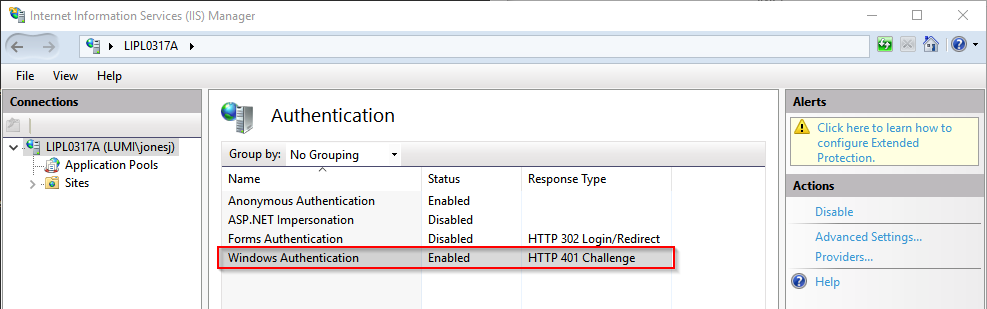
1. You do not need to reset IIS after changing the web.config!

\*\*\*See next page\*\*\*

1. Windows authentication must be on in IIS. To check, click on ‘Default web site’ (or the relevant site).
2. Click on ‘Authentication as shown below.’

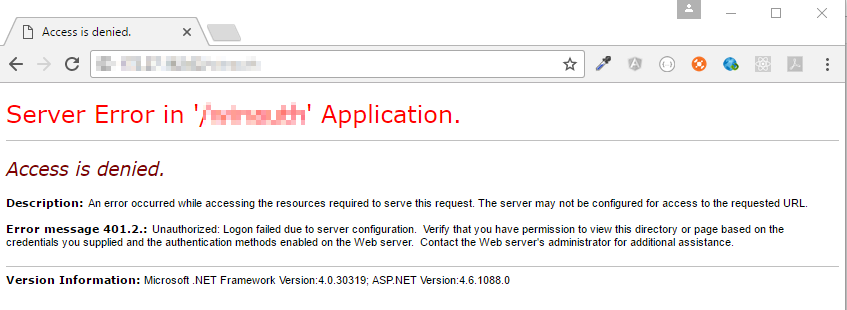


1. Set the ‘Windows Authentication’ to enabled.

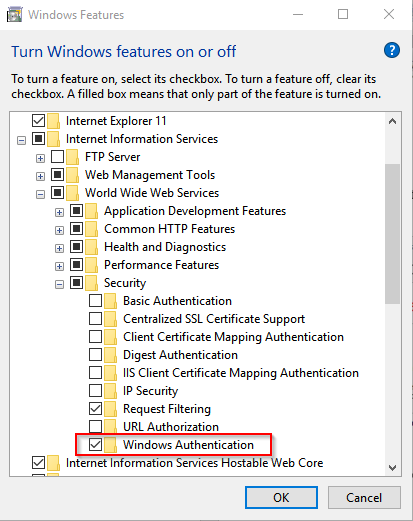


*(see the next point if you don’t see this setting)*

If it’s not, you get the following error:



1. If the above setting isn’t visible in IIS, do the following:
   1. Go to ‘Turn Windows Features On or Off’.
   2. Check Internet Information Services > World Wide Web Services > Security > Windows Authentication.



1. To see the logged in user, use @User.Identity.Name in a razor view.
2. If you want to have a logout, the following will enable it

/// <summary>

/// http://stackoverflow.com/questions/17871816/login-as-another-user-mvc-4-windows-authentication

/// </summary>

/// <returns></returns>

[AllowAnonymous]

public ActionResult LogOut()

{

HttpCookie cookie = Request.Cookies["TSWA-Last-User"];

if (User.Identity.IsAuthenticated == false || cookie == null || StringComparer.OrdinalIgnoreCase.Equals(User.Identity.Name, cookie.Value))

{

string name = string.Empty;

if (Request.IsAuthenticated)

{

name = User.Identity.Name;

}

cookie = new HttpCookie("TSWA-Last-User", name);

Response.Cookies.Set(cookie);

Response.AppendHeader("Connection", "close");

Response.StatusCode = 401; // Unauthorized;

Response.Clear();

//should probably do a redirect here to the unauthorized/failed login page

//if you know how to do this, please tap it on the comments below

Response.Write("Unauthorized. Reload the page to try again...");

Response.End();

return RedirectToAction("Index");

}

cookie = new HttpCookie("TSWA-Last-User", string.Empty)

{

Expires = DateTime.Now.AddYears(-5)

};

Response.Cookies.Set(cookie);

return RedirectToAction("Index");

}

1. END