

RE: FormsAuthentication returnUrl – need it to be Absolute

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.aspnet/2006-09/msg02893.html>

- *From:* Noremac <Noremac@xxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Thu, 21 Sep 2006 06:04:03 -0700
-

Thanks Steven. Extremely helpful as always.

"Steven Cheng[MSFT]" wrote:

Hello Noremac,

From your description, I understand you have multiple ASP.NET applications which are using forms authentication to protect the application. Currently you're trying to make the applications share the same forms authentication cookie/ticket so as to make the users in those applications single-signing(SSO), correct?

Based on my experience, though the ASP.NET application does support share the forms authentication cookie across multiple applications, there has many limitation on this. In addition to those forms authentication and encryption key setting mentioned in the following article:

#Forms Authentication Across Applications
<http://msdn2.microsoft.com/en-us/library/eb0zx8fc.aspx>

You need to make sure that those different ASP.NET applications which want to share the forms authentication cookie must be hosted on servers which are accessed under the same top level domain. e.g.

the following two application can share cookie:

<http://subxxx.mymaindomain.com/>

<http://mymaindomain.com/>

while the below two can not share cookie

<http://mydomain1.com/>

<http://mydomain2.com>

RE: FormsAuthentication returnUrl – need it to be Absolute

Therefore, you need to make sure the applications in your environment meet the above requirements.

As for pass the returnUrl(the original url visited) to the login form, the cookie approach does be workable. I've tested it in my local test environment. Actually, you need to add the cookie into the `HttpResponse.Cookies` collection(rather than `Request.cookies`). Also, you need to use a different cookiename from the `FormsAuthentication.FormsCookiePath`. And make sure the Cookie's `DomainName` and path is identical to the formsauthentication's settign. Here is my test code which works well in my local test.

```
protected void Application_AuthenticateRequest(object sender, EventArgs e)
{
    if (!Request.IsAuthenticated)
    {
        HttpCookie formCookie = new
        HttpCookie(FormsAuthentication.FormsCookieName + "redirecturl");

        formCookie.Domain = FormsAuthentication.CookieDomain;
        formCookie.Path = FormsAuthentication.FormsCookiePath;

        formCookie.Value = Request.Url.AbsoluteUri;

        Response.Cookies.Add(formCookie);
    }
}
```

=====

If the forms authentication cross site support doesn't suit your scenario. I'm afraid you may need to consider implementing your custom SSO mechanism. In addition, here are some other articles discussing on this topic.

<http://weblogs.asp.net/scottgu/archive/2005/12/10/432851.aspx>

<http://www.codeproject.com/aspnet/aspnetsinglesignon.asp>

Hope this helps. If there is anything unclear on this, please feel free to let me know.

Sincerely,

Steven Cheng

Microsoft MSDN Online Support Lead

=====

Get notification to my posts through email? Please refer to <http://msdn.microsoft.com/subscriptions/managednewsgroups/default.aspx#notifications>.

Note: The MSDN Managed Newsgroup support offering is for non-urgent issues where an initial response from the community or a Microsoft Support Engineer within 1 business day is acceptable. Please note that each follow up response may take approximately 2 business days as the support professional working with you may need further investigation to reach the most efficient resolution. The offering is not appropriate for situations that require urgent, real-time or phone-based interactions or complex project analysis and dump analysis issues. Issues of this nature are best handled working with a dedicated Microsoft Support Engineer by contacting Microsoft Customer Support Services (CSS) at <http://msdn.microsoft.com/subscriptions/support/default.aspx>.

=====

This posting is provided "AS IS" with no warranties, and confers no rights.