

Re: SignOut a User

Source:

<http://www.derkeiler.com/Newsgroups/microsoft.public.dotnet.framework.aspnet.security/2002-12/2154.html>

From: Mike Moore [MS] (michmo@online.microsoft.com)

Date: 12/11/02

From: michmo@online.microsoft.com ("Mike Moore [MS]")

Date: Wed, 11 Dec 2002 05:19:34 GMT

Hi Paps,

I agree with Yariv. I also think you should reconsider if this is desirable. I do not know your individual situation. So, please don't take the following as the only answer.

Suppose I logon to a machine at my desk and start some work. Then I go to a conference at another building within my company. While I'm there, I realize I should have done something. I logon to a machine and get it done. Later, when I return to my desk, I expect to continue where I left off, not find that my authentication has been revoked.

Now, assuming that you do need to revoke "older" authenticated sessions.... As part of the logon process, first verify that this person is allowed. Then, instead of calling RedirectFromLoginPage, call GetRedirectUrl. Next, lookup this user in a database which lists users and their last known authentication cookie value. If there is already an entry, update it with the new value. If there is no entry, add one. Then you can use Response.Redirect to the return value you already got from GetRedirectUrl. Next, add code, that runs with every viewing of every page, that compares the current user's name and authentication cookie with the data in the database and deny access if there is no match. This will deny the older session if the same user starts a second session.

Try the BeginRequest event in the global.asax for running this test against the database. I did not try accessing a database from this event, but I did verify that Request.Cookies can be accessed in this event. Not all objects are available in this event. The session object and request.Form are not available.

You can access the authentication cookie in the BeginRequest event as follows:

```
Try
    Auth =
Request.Cookies(System.Web.Security.FormsAuthentication.FormsCookieName).Value
Catch
End Try
```

Does this answer your question?

Thank you, Mike Moore

Microsoft, ASP.NET

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

Re: SignOut a User

microsoft.public.dotnet.framework.aspnet.security: Re: SignOut a User

>From: "Y" <noReply@dot.net>
>References: <OPy7CV5nCHA.2288@TK2MSFTNGP10>
<tSHfvU\$nCHA.2044@cpmsftngxa06> <eiLl#hFoCHA.2312@TK2MSFTNGP09>
>Subject: Re: SignOut a User
>Date: Tue, 10 Dec 2002 22:38:44 +0200
>Lines: 146
>X-Priority: 3
>X-MSMail-Priority: Normal
>X-Newsreader: Microsoft Outlook Express 6.00.2600.0000
>X-MimeOLE: Produced By Microsoft MimeOLE V6.00.2600.0000
>Message-ID: <OdD2ewIoCHA.2404@TK2MSFTNGP10>
>Newsgroups: microsoft.public.dotnet.framework.aspnet.security
>NNTP-Posting-Host: ras3-p88.rvt.netvision.net.il 62.0.182.88
>Path: cpmsftngxa09!cpmsftngxa10!tkmsftngp01!TK2MSFTNGP10
>Xref: cpmsftngxa09 microsoft.public.dotnet.framework.aspnet.security:3311
>X-Tomcat-NG: microsoft.public.dotnet.framework.aspnet.security
>
>as far as I know it doesn't matter how many connections there are, nor
>their
>locations.
>the cookie (after successful login) is created per browser.
>so if your a user login from 2 different browsers (locations) 2 different
>cookies are created. So that user will have to sign-out twice.
>
>in case you want to close all it's connections at once, you'll have to
>build
>a server-state-machine that will handle the user state.
>
>Yariv,
><mailto:Creamelana@myrealbox.com>
>
>"Ppaps" <papettip@euroforex.com> wrote in message
>news:eiLl#hFoCHA.2312@TK2MSFTNGP09...
>> Thanks Mike for your reply.
>>
>> Q:
>> I understood your sample code, but if the same user is connecting at
>the
>> same time from 2 different workstation I would like that that the oldest
>> connection is signed out.
>>
>> > Code for webform1.aspx
>> > Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
>> > System.EventArgs) Handles Button1.Click
>> > FormsAuthentication.SignOut()
>> > Response.Redirect("webform1.aspx")
>> > End Sub
>> >
>>
>>
>> If I call FormsAuthentication.SignOut() I'm calling the signout of the
>> newest connection and not the old one.
>> So I need during the Newest-user-authentication retrieve the oldest one
>> and
>> Sign it Out.
>>
>> (forgive me 4 loosing time :-))
>> Thanks Paps
>>
>>
>> "Mike Moore [MS]" <michmo@online.microsoft.com> wrote in message

microsoft.public.dotnet.framework.aspnet.security: Re: SignOut a User

```
>> news:tSHfvU$nCHA.2044@cpmsftngxa06...
>> > Hi Paps,
>> >
>> > I just tried several experiments with FormsAuthentication.SignOut and
>> > RedirectFromLoginPage to double check their behavior.
>> >
>> > When you call FormsAuthentication.SignOut, the cookie that represents
>> > the
>> > previously authenticated user is removed.
>> >
>> > Code for webform1.aspx
>> >     Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
>> > System.EventArgs) Handles Button1.Click
>> >         FormsAuthentication.SignOut()
>> >         Response.Redirect("webform1.aspx")
>> >     End Sub
>> >
>> > This button event signs out and redirects. The signout clears the
>> > cookie.
>> > However, if the redirect was not included, then the same page the user
>> > was
>> > already viewing will be displayed again. Authorization takes place very
>> > early, before the page is processed. By the time this code removes the
>> > authorizing cookie, the page is already being processed. So, the page
>> > gets
>> > redisplayed after the call to signout. Though the user can still view
>> > the
>> > page as it appears on the screen, they cannot do anything more. Any
>> > activity will redirect them to the login page. Having the page redirect
>> > to
>> > itself, or any appropriate page, brings up the login page.
>> >
>> > If you redirect directly to the login page, then the user will be at
>> > the
>> > login page with nothing in the query string to indicate where to go to
>> > next. My experiments showed that
>> > FormsAuthentication.RedirectFromLoginPage
>> > would redirect to "default.aspx" -- even if that page is not listed
>> > as
>> > a
>> > default document in IIS and even if that page does not exist.
>> >
>> > Code for login.aspx
>> >     Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
>> > System.EventArgs) Handles Button1.Click
>> >         'commented out: FormsAuthentication.SignOut()
>> >         FormsAuthentication.RedirectFromLoginPage(TextBox1.Text, True)
>> >     End Sub
>> >
>> > I also experimented with and without the call to SignOut just prior to
>> > RedirectFromLoginPage. I didn't find any difference. Using:
>> >     Response.Write("<br>User: " & User.Identity.Name)
>> > showed the new user regardless of the call to SignOut.
>> >
>> > So, I found that SignOut is useful for removing an authorization if you
>> > are
>> > not also granting a new one at the same time. But, it was not useful if
>> > you
>> > are about to call RedirectFromLoginPage. The new call to
>> > RedirectFromLoginPage would update the authentication anyway.
>> >
>> > Does this answer your question?
```

microsoft.public.dotnet.framework.aspnet.security: Re: SignOut a User

```
>> >
>> > Thank you, Mike Moore
>> > Microsoft, ASP.NET
>> >
>> > This posting is provided "AS IS", with no warranties, and confers no
>> > rights.
>> >
>> > -----
>> > >From: "Ppaps" <papettip@euroforex.com>
>> > >Subject: SignOut a User
>> > >Date: Mon, 9 Dec 2002 16:10:53 +0100
>> > >Lines: 14
>> > >X-Priority: 3
>> > >X-MSMail-Priority: Normal
>> > >X-Newsreader: Microsoft Outlook Express 6.00.2800.1106
>> > >X-MimeOLE: Produced By Microsoft MimeOLE V6.00.2800.1106
>> > >Message-ID: <OPy7CV5nCHA.2288@TK2MSFTNGP10>
>> > >Newsgroups: microsoft.public.dotnet.framework.aspnet.security
>> > >NNTP-Posting-Host: 151.99.166.162
>> > >Path: cpmsftngxa09!cpmsftngxa10!tkmsftngp01!TK2MSFTNGP10
>> > >Xref: cpmsftngxa09
>microsoft.public.dotnet.framework.aspnet.security:3294
>> > >X-Tomcat-NG: microsoft.public.dotnet.framework.aspnet.security
>> > >
>> > >I use Authentications Forms to authenticcate my users, but , if a user
is
>> > >connecting with a login tha is already connected, I need to SignOut
the
>> > >old
>> > >>connected user.
>> > >>ex
>> > >>if (IsConnected(User.Identity.UserName)
>> > >>{
>> > >>    OldUserFormsAuthentication.SignOut(); <--- something like this....
>> > >>}
>> > >
>> > >>How can I do this ?
>> > >
>> > >>Thanks Paps
>> > >
>> > >
>> > >
>> >
>>
>
>
>
```