

Re: Where is the user impersonation token stored?

Source:

<http://www.derkeiler.com/Newsgroups/microsoft.public.dotnet.framework.aspnet.security/2005-10/0096.html>

From: Gery D. Dorazio (*gdorazio_at_enqueue.net*)

Date: 10/11/05

Date: Mon, 10 Oct 2005 20:50:57 -0400

Dominick,

It looks like this is an IIS question now. I am going to look around over on the IIS.Security newsgroup and possibly post over there also.

Thanks,
Gery

--

Gery D. Dorazio
Development Engineer
EnQue Corporation
www.EnQue.com
www.ImagingHardware.com
"Dominick Baier [DevelopMentor]" <dbaier@pleasepleasenosspamdevelop.com>
wrote in message <news:425654601063cc8c79c1f74e7fcdd@news.microsoft.com>...
> Hello Gery,
>
> first of all the browser never tries to authenticate if he doesn't need
> to - when you hit a page where you don't have anonymous access - IIS
> bounces back a 401 to the client along with the possible authentication
> methods.
>
> From that point on the browser sends the authentication information as a
> header to the server on each request.
>
> download www.fiddlertool.com and try the different authentication settings
> in IIS - you can see the headers flowing back and forth. A good testpage
> to test the various settings can be found here:
> <http://www.leastprivilege.com/UpdatedShowContextsAndRequestLogonUserIdentity.aspx>
>
> -----
> Dominick Baier - DevelopMentor
> <http://www.leastprivilege.com>
>
>> Hi Dominick,
>>
>> Thanks for the feedback. Can you explain a little more with respect to
>> IIS?
>>
>> Here is the scenareo that has me stumped and really the reason for the
>> post:
>>
>> 1) User requests a restricted page and the Windows popup dialog

microsoft.public.dotnet.framework.aspnet.security: Re: Where is the user impersonation token stored?

```
>> appears so
>> the user logs in and is authenticated. Then the page is served up.
>> 2) The user then clicks on another secured page link and is directed
>> to that
>> page...no popup since he is already authenticated.
>> Here is a question that may hit the core of the problem:
>>
>> How does IIS handle authenticated Windows accounts during
>> client-server requests to a web server?
>>
>> Here is my thinking as to what happens and the source of my confusion:
>>
>> When an HTTP request is finished and the response is sent back to the
>> client the worker thread is finished and recycled...at least that's
>> how I understand it. Along with this understanding the server then
>> would have no knowledge whether the user is logged in...eg http is
>> stateless. Then for subsequent requests IIS would have to log them in
>> automatically for each request since they already logged in once. But
>> what happens on the next request? Where does IIS (or some ISAPI
>> authentication filter/extension) get the information to re-logon the
>> user? Translated...where is this: ctx.WorkerRequest.GetUserToken()
>> getting its user token from?...is it stored in a header, an encrypted
>> cookie passed back and forth between client and server?...all this is
>> only in regards to Windows authentication and not ASP.NET forms
>> authentication since I know that is encrypted in a forms cookie...
>>
>> Thanks and hope this is clear,
>>
>> Thanks,
>> Gery
>> EnQue Corporation
>> www.EnQue.com
>> www.ImagingHardware.com
>> "Dominick Baier [DevelopMentor]"
>> <dbaier@pleasepleasenosspamdevelop.com> wrote in message
>> news:42565460104b138c79b9ca7e77d13@news.microsoft.com...
>>
>>> Hello Gery,
>>>
>>> 1) The outcome os IIS authentication is stored in a blob called ISAP
>>> Extension Control Block - the ASPNET_ISAPI extension passes the token
>>> to ASP.NET (via WorkerRequest). This token is availabe in ASP.NET 2.0
>>> using the Request.LogonUserIdentity
>>>
>>> 2) There is some caching involved in IIS - but ASP.NET grabs the
>>> impersonation token on each request from IIS to populate
>>> Context.User.
>>>
>>> HTH
>>> -----
>>> Dominick Baier - DevelopMentor
>>> http://www.leastprivilege.com
>>>> When a user visits a web site and is authenticated through the popup
>>>> dialog box (Windows authentication) he enters his username and
>>>> password. Evidently this creates the users impersonation token that
>>>> is used on subsequent requests to secured web pages. On subsequent
>>>> requests the WindowsAuthenticationModule is what authenticates on
>>>> each request. The code that does this looks like this:
>>>>
>>>> WindowsIdentity wi = new
>>>> WindowsIdentity(ctx.WorkerRequest.GetUserToken(),
>>>> text2, WindowsAccountType.Normal, true);
```

microsoft.public.dotnet.framework.aspnet.security: Re: Where is the user impersonation token stored?

```
>>>> Context.User = new WindowsPrincipal(wi);
>>>> The questions are:
>>>> 1. Where did the initial Windows authentication put the user
>>>> impersonation
>>>> token?
>>>> 2. Where is the user impersonation token stored as the user makes
>>>> web
>>>> page
>>>> requests(or is it generated on each request and if so how)?
>>>> Thanks,
>>>> Gery
>>>> EnQue Corporation
>>>> www.EnQue.com
>>>> www.ImagingHardware.com
>
>
```