

Re: Propagating caller identity across applications from a bare ASMX Service method to a WSE3 Service method

Re: Propagating caller identity across applications from a bare ASMX Service method to a WSE3 Service method

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.webservices.enhancements/2007-01/>

- *From:* "Howard Hoffman" <HowardH@xxxxxxxxxxxxxxxxxxxx>
 - *Date:* Wed, 10 Jan 2007 10:24:22 -0500
-

Steven –

Thanks for the quick reply. I'm familiar with the articles you mention, but I'll re-read them to see if I've missed something (probably have missed a few things).

In production, I will be able to have the Proxy Web service on the same computer as the End Web Service. We control that configuration. Our understanding is that all the client accounts will be in the same Active Directory Domain as the server computer and the server App Pool run-as account.

Unfortunately, the 'edge' web application is not controlled by us -- its a customer authored application -- and it sits on a different computer. That computer and its App Pool run-as account (which is probably Network Service) are not controlled by us.

I'll have to check whether the customer is running Active Directory in Windows 2003 Server mode -- they may be in Windows 2000 mixed mode.

Howard

"Steven Cheng[MSFT]" <stcheng@xxxxxxxxxxxxxxxxxxxx> wrote in message news:F7E%23WYINHHA.2488@xxxxxxxxxxxxxxxxxxxx

Hello Howard,

Glad to see you again.

Seems you're still struggling with the multi-tier webservice authentication issue. As you described here, your original architecture is

client browser <---> web application <---> end webservice (windows authentication)

Re: Propagating caller identity across applications from a bare ASMX Service method to a WSE3 Service method

Re: Propagating caller identity across applications from a bare ASMX Service method to a WSE3 Service method

and currently, due to some asp.net web application is using .NET 1.1 (can not afford wse 3.0). You adjust the architecture as below:

client browser <---> web application<--->proxy webservice<--->end webservice (windows authentication).

I think forwarding the windows security identity from client browser to the end webservice (across all the intermediate services) is quite difficult. To do this, all the computers (from client to end services and intermediate server) should be in the same local domain or trusted domain. And they need to be configured so as to use kerberos delegation. Configuring kerberos delegation including configuring all the possible client accounts and all the server account and machines which will perform delegation. Just the same as configuring double hop scenario for ASP.NET application.

#How to configure an ASP.NET application for a delegation scenario
<http://support.microsoft.com/kb/810572/en-us>

Have all the application and services in your scenario been configured for kerberos delegation? For configuration specific problem, there is a document for troubleshooting kerberos delegation

#Kerberos authentication and troubleshooting delegation issues
<http://support.microsoft.com/kb/907272/en-us>

And as you mentioned, currently you put all the services on the server machine, is this also the topology in production environment? If not, I suggest you make other services on a separate machine with the ASP.NET web application. Because if they're on the same machine, the windows identity in ASP.NET web application can be forwarded to other services on the same machine without using kerberos delegation.

In addition, I'm not sure whether using kerberos delegation and windows authentication across all the hops is a must-to-use approach for your scenario. For windows 2003 domain, there has provided a constrained delegation and S4U type token that can help the server generate a windows security token (the windowsIdentity class in .net framework) without password (only the user principal name is necessary). Thus, it opens a way for N-TIER application which need flow security identity/context across multi application/services. Here are some MSDN reference describing on this, I think this is also an option you can consider.

#How To: Use Protocol Transition and Constrained Delegation in ASP.NET 2.0
<http://msdn2.microsoft.com/en-us/library/ms998355.aspx>

#Exploring S4U Kerberos Extensions in Windows Server 2003

Re: Propagating caller identity across applications from a bare ASMX Service method to a WSE3 Service method

Re: Propagating caller identity across applications from a bare ASMX Service method to a WSE3 Service method

<http://msdn.microsoft.com/msdnmag/issues/03/04/SecurityBriefs/default.aspx>

Hope this helps.

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.

Re: Propagating caller identity across applications from a bare ASMX Service method to a WSE3 Service method

Re: Propagating caller identity across applications from a bare ASMX Service method to a WSE3 Service method