**Use Windows Azure Access Control Service with JWT(CS\VBAzureACSJWT)**

**Introduction**

Windows Azure Access Control Service integrates WIF, so ASP.NET developers can easily create Claims-Aware Application by Identity and Access extension. But for C/S application, developers can’t  add STS reference to their client, it’s harder to use ACS with client application and web service.

This article and the attached code samples demonstrate how to use Azure ACS work with third part Identity provider such as google, yahoo. You can find the answers for all the following questions in the code sample:

How to use third part IDP such as google, yahoo in WPF.

How to get RP's claims information in WPF client app.

How to desterilize security token provided by google or yahoo.

**Running the Sample**

You should do the steps below before running the code sample.

Step 1: To configure the REST web service as a relying party

1. Go to the [Windows Azure Management Portal](http://go.microsoft.com/fwlink/p/?LinkID=275081), sign in, and then click **Active Directory**.
2. To manage an Access Control namespace, select the namespace, and then click **Manage**. (Or, click **Access Control Namespaces**, select the namespace, and then click **Manage**.)
3. In the **Trust Relationships** section, click **Relying Party Applications**.
4. On the **Relying Party Applications** page, click **Add link**. The **Add Relying Party Application** page opens.
5. In the **Relying Party Application Settings** section, make the following selections:
   * **Name**—Specify a display name for this relying party, for example, CSAzureACSAuthInWPF.
   * **Mode**—Select the **Enter settings manually** option.
   * **Realm**—Specify the realm of your WCF service, for example, <http://localhost:12526/RESTUserService.svc>**.**
   * **Return URL**—Leave blank.
   * **Error URL**—Leave blank.
   * **Token format**—Select the **SWT** option.
   * **Token lifetime (secs)**—Leave the default of 600 seconds.
6. In the **Authentication Settings** section, make the following selections:
   * **Identity providers**—checke google and yahoo.
   * **Rule groups**—Select the **Create New Rule Group** option.
7. In the **Token Signing Settings** section, make the following selections:
   * **Token signing**—Select the **Use a dedicated certificate** option.
   * **Token signing key**—To generate 256–bit symmetric key, click **Generate**.
   * **Effective date**—specify the key’s effective date.
   * **Expiration date**—specify the key’s expiration date.
8. Click **Save**.

Saving your project will also trigger the creation of a rule group. Now you need to add rules in the rule group.

Step 2: Change parameters to your own in below files.

1. CSAzureACSJWT\ App.xaml.cs file.

**Using the Code**

The code sample provides the following functions to resolve the questions above.

**How to use third part IDP such as google, yahoo in WPF.**

|  |
| --- |
| -Code block start-  --C# code snippet start--  private void GetIdentityProviders()  {  {  Uri identityProviderDiscovery = new Uri(  string.Format(CultureInfo.InvariantCulture,  "https://{0}.{1}/v2/metadata/IdentityProviders.js?protocol=javascriptnotify&realm={2}&version=1.0",  App.serviceNamespace,  App.acsHostUrl,  HttpUtility.UrlEncode(App.realm)),  UriKind.Absolute  );  WebClient webClient = new WebClient();  webClient.DownloadStringCompleted += new DownloadStringCompletedEventHandler(WebClientDownloadStringCompleted);  webClient.DownloadStringAsync(identityProviderDiscovery);  }  }  --C# code snippet end--  --VB code snippet start--  Private Sub GetIdentityProviders()  If True Then  Dim identityProviderDiscovery As New Uri(String.Format(CultureInfo.InvariantCulture, "https://{0}.{1}/v2/metadata/IdentityProviders.js?protocol=javascriptnotify&realm={2}&version=1.0", Application.serviceNamespace, Application.acsHostUrl, HttpUtility.UrlEncode(Application.realm)), UriKind.Absolute)  Dim webClient As New WebClient()  AddHandler webClient.DownloadStringCompleted, AddressOf WebClientDownloadStringCompleted  webClient.DownloadStringAsync(identityProviderDiscovery)  End If  End Sub  --VB code snippet end--  -Code block end- |

**How to get RP's claims information in WPF client app.**

|  |
| --- |
| -Code block start-  --C# code snippet start--  [System.Runtime.InteropServices.ComVisibleAttribute(true)]  public class HtmlInteropClass  {  public void Notify(string jsonToken)  {  var jwtSTH = getDeserializedToken(jsonToken);  foreach (var claim in jwtSTH.Claims)  {  if (claim.Type == "<http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress>")  {  Settings.Default.CustomerEmail = claim.Value;  }  ((MainWindow)Application.Current.MainWindow).tblMessage.Text += claim.ToString() + "\n";  }  foreach (var window in Application.Current.Windows)  {  if (window as Login != null)  {  ((Login)window).Close();  ((MainWindow)Application.Current.MainWindow).stateCheck();  }  }  }  /// <summary>  /// Third part IDP provider will provide issure a Json formate token, and serialized JWT in "securityToken".  /// This method will deserialized the Json token and return JwtSecurityToken.  /// </summary>  /// <param name="jsonToken"></param>  /// <returns></returns>  private JwtSecurityToken getDeserializedToken(string jsonToken)  {  dynamic jObj = JsonConvert.DeserializeObject(jsonToken);  var securityTokenValue = jObj["securityToken"].ToString();  JwtSecurityTokenHandler jwtSTH = new JwtSecurityTokenHandler();  var jwtST = jwtSTH.ReadToken(securityTokenValue) as JwtSecurityToken;  return jwtST;  }  }  --C# code snippet end--  --VB code snippet start--  <System.Runtime.InteropServices.ComVisibleAttribute(True)> \_  Public Class HtmlInteropClass  Public Sub Notify(jsonToken As String)  Dim jwtSTH = getDeserializedToken(jsonToken)  For Each claim In jwtSTH.Claims  If claim.Type = "<http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress>" Then  My.Settings.CustomerEmail = claim.Value  End If  DirectCast(Application.Current.MainWindow, MainWindow).tblMessage.Text += claim.ToString() + vbLf  Next  For Each window In Application.Current.Windows  If TryCast(window, Login) IsNot Nothing Then  DirectCast(window, Login).Close()  DirectCast(Application.Current.MainWindow, MainWindow).stateCheck()  End If  Next  End Sub  ''' <summary>  ''' Third part IDP provider will provide issure a Json formate token, and serialized JWT in "securityToken".  ''' This method will deserialized the Json token and return JwtSecurityToken.  ''' </summary>  ''' <param name="jsonToken"></param>  ''' <returns></returns>  Private Function getDeserializedToken(jsonToken As String) As JwtSecurityToken  Dim jObj As Object = JsonConvert.DeserializeObject(jsonToken)  Dim securityTokenValue = jObj("securityToken").ToString()  Dim jwtSTH As New JwtSecurityTokenHandler()  Dim jwtST = TryCast(jwtSTH.ReadToken(securityTokenValue), JwtSecurityToken)  Return jwtST  End Function  End Class  --VB code snippet end--  -Code block end- |

**How to desterilize security token provided by google or yahoo.**

|  |
| --- |
| -Code block start-  --VB code snippet start--  /// <summary>  /// Third part IDP provider will provide issure a Json formate token, and serialized JWT in "securityToken".  /// This method will deserialized the Json token and return JwtSecurityToken.  /// </summary>  /// <param name="jsonToken"></param>  /// <returns></returns>  private JwtSecurityToken getDeserializedToken(string jsonToken)  {  dynamic jObj = JsonConvert.DeserializeObject(jsonToken);  var securityTokenValue = jObj["securityToken"].ToString();  JwtSecurityTokenHandler jwtSTH = new JwtSecurityTokenHandler();  var jwtST = jwtSTH.ReadToken(securityTokenValue) as JwtSecurityToken;  return jwtST;  }  --VB code snippet end--  --VB code snippet start--  ''' <summary>  ''' Third part IDP provider will provide issure a Json formate token, and serialized JWT in "securityToken".  ''' This method will deserialized the Json token and return JwtSecurityToken.  ''' </summary>  ''' <param name="jsonToken"></param>  ''' <returns></returns>  Private Function getDeserializedToken(jsonToken As String) As JwtSecurityToken  Dim jObj As Object = JsonConvert.DeserializeObject(jsonToken)  Dim securityTokenValue = jObj("securityToken").ToString()  Dim jwtSTH As New JwtSecurityTokenHandler()  Dim jwtST = TryCast(jwtSTH.ReadToken(securityTokenValue), JwtSecurityToken)  Return jwtST  End Function  End Class  --VB code snippet end--  -Code block end- |