

Defect Details

Defect Id: 223

Project: Gallery Server Pro

Name: (SQLite) Role and profile providers must use same application name as membership provider

Date Found: 6/23/2009

Build #: 2.3.3456

Severity: Low Impact

Date Fixed: 7/2/2009

Build # of Fix: 2.3.3470

Status: Closed

Estimated Work: 4 hrs

Assigned To: Roger Martin

Priority: Low

Actual Work: 3 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

It is possible to configure GSP to use a different application for roles and profiles than users, like this:

```
<membership defaultProvider="SqlMembershipProvider">
<providers>
<clear />
<add applicationName="YAF" name="SqlMembershipProvider" ... />
</providers>
</membership>

<roleManager defaultProvider="SqlRoleProvider" ...>
<providers>
<clear />
<add applicationName="Gallery Server Pro" name="SqlRoleProvider" ... />
</providers>
</roleManager>

<profile defaultProvider="SqlProfileProvider">
<providers>
<clear />
<add applicationName="Gallery Server Pro" name="SqlProfileProvider" ... />
</providers>
...
</profile>
```

This allows for the possibility of, for example, integrating GSP into a web application using YetAnotherForum (YAF) where both GSP and YAF use a single user store but different sets of roles.

This technique works in SQL Server but not in SQLite.

In SQLite, it is impossible to add a user to a role, since the SQL that links the two assumes the same application ID for both. For example, refer to the `SQLiteRoleProvider.AddUsersToRoles` method. The SQL Server provider gets around this by listing each user one time in `aspnet_Membership` (as you would expect) but duplicating them in `aspnet_Users`, once for each application. SQLite does not have two tables - just a single `aspnet_Users` - so this may be tricky to fix. Perhaps need to add a column that marks a record as the "master" to distinguish usernames in other applications that aren't really associated with them. Or perhaps update the SQL so that the application ID for users and roles is kept separate.

Resolution

The `SQLiteRoleProvider` and `SQLiteProfileProvider` classes were modified so that the application ID associated

Defect Details

with the membership is used when querying the aspnet_Users table and the application ID associated with roles is used with the aspnet_Roles table. Previously the code assumed the users and roles were both associated with the roles' application ID.

Since the roles and profile providers have no knowledge of what the application ID is for the Membership, a property named **MembershipApplicationName** was added to each class. This property can be specified in web.config, as seen here for roles:

```
<
roleManagerenabled="true"cacheRolesInCookie="true"cookieProtection="Validation"defaultProvider="SQLiteRoleProvider">
<providers>
<clear/>
<addapplicationName="GSP"membershipApplicationName="YAF"connectionStringName="SQLiteDbConnection"name="
SQLiteRoleProvider"type="GalleryServerPro.Data.SQLite.SQLiteRoleProvider"/>
</providers>
```

</roleManager>The membershipApplicationName attribute is optional. When not specified or empty, it assumes the Membership has the same application name as the roles, which will be true in most cases.

I did not choose the technique used by the writers of the SqlMembershipProvider, which requires two user-related tables (aspnet_Users and aspnet_Membership) and two records in aspnet_Users for each user when the application name is different between the membership and roles provider definitions. Besides having the benefit of not changing the data schema, it also seems cleaner.

Defect Id: 224

Project: Gallery Server Pro

Name: Disabling a user album through the Manage Users page does not delete the album

Date Found: 6/24/2009

Build #: 2.3.3456

Severity: Low Impact

Date Fixed: 7/2/2009

Build # of Fix: 2.3.3470

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: Low

Actual Work: 30 min

Reported By: Roger Martin

Percent Complete: 100

Description

Disabling a user album through the Manage Users page does not delete the album. This is in contrast to a user disabling their user album on their account page, which does delete the album.

The album should be deleted, regardless of where the user album is disabled.

Resolution

There was an incorrect sequence of events in the SaveProfile function in Website\gs\pages\admin\manageusers.ascx.cs. The call to ProfileController.SaveProfile(userProfile) must happen at the end of the function, not before the user album deletion. Once this was changed, the user album was properly deleted.

Defect Details

Defect Id: 225

Project: Gallery Server Pro

Name: Cannot approve user with Internet Explorer

Date Found: 6/29/2009

Build #: 2.3.3456

Severity: Medium Impact

Date Fixed: 6/29/2009

Build # of Fix: 2.3.3467

Status: Closed

Estimated Work: 2 hrs

Assigned To: Roger Martin

Priority: Medium

Actual Work: 2 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

When the "Require administrator approval" option is enabled, administrators are sent an e-mail when a user must be approved. This e-mail contains a link that opens the edit user dialog for the new user on the Manage Users page. In Internet Explorer, a javascript error occurs within one of the ComponentArt javascript files because the dialog is anchored to the Edit link for the user in the grid, but the grid has not completed loading.

Resolution

I added a client event handler to the Grid control that set a global variable gdUsersLoaded to true when the grid has completed loading. Then, in the function editUser I check this variable before assigning an HTML element to the dialog's AnimationDirectionElement property as follows:

```
function editUser(userName)
{
  ...
  if (gdUsersLoaded)
    dgEditUser.set_animationDirectionElement(makeValidForId(userName));
  else
    dgEditUser.set_animationDirectionElement('addnewuser');
  ...
}
```

The end result is that if the grid hasn't completed loading by the time the editUser function is running, we set the AnimationDirectionElement property to the "addnewuser" HTML element, which is guaranteed to exist.

Defect Id: 226

Project: Gallery Server Pro

Name: (SQL Server) SQL error after upgrading from 2.0 - 2.2

Date Found: 6/29/2009

Build #: 2.3.3456

Severity: Medium Impact

Date Fixed: 6/29/2009

Build # of Fix: 2.3.3467

Status: Closed

Estimated Work: 10 min

Assigned To: Roger Martin

Priority: High

Actual Work: 10 min

Reported By: Roger Martin

Percent Complete: 100

Description

When using SQL Server and upgrading from 2.0 - 2.2, the following error may occur:

Error: The type initializer for 'GalleryServerPro.Web.Pages.GalleryPage' threw an exception.

Exception Type: System.TypeInitializationException

Defect Details

Message: The type initializer for 'GalleryServerPro.Web.Pages.GalleryPage' threw an exception.

Source: GalleryServerPro.Web

Stack Trace

```
at GalleryServerPro.Web.Pages.GalleryPage..ctor()
at GalleryServerPro.Web.gs.pages.album..ctor()
at ASP.gs_pages_album_ascx..ctor()
at __ASP.FastObjectFactory_app_web_1qegexqw.Create_ASP_gs_pages_album_ascx()
at System.Web.Compilation.BuildResultCompiledType.CreateInstance()
at System.Web.UI.TemplateControl.LoadControl(IWebObjectFactory objectFactory, VirtualPath virtualPath, Type t, Object[] parameters)
at System.Web.UI.TemplateControl.LoadControl(VirtualPath virtualPath)
at System.Web.UI.TemplateControl.LoadControl(String virtualPath)
at GalleryServerPro.Web.Gallery.LoadRequestedPage()
at GalleryServerPro.Web.Gallery.Gallery_Init(Object sender, EventArgs e)
at System.Web.UI.Control.OnInit(EventArgs e)
at System.Web.UI.UserControl.OnInit(EventArgs e)
at System.Web.UI.Control.InitRecursive(Control namingContainer)
at System.Web.UI.Control.InitRecursive(Control namingContainer)
at System.Web.UI.Control.InitRecursive(Control namingContainer)
at System.Web.UI.Page.ProcessRequestMain(Boolean includeStagesBeforeAsyncPoint, Boolean includeStagesAfterAsyncPoint)
```

Inner Ex Type: System.Data.SqlClient.SqlException

Inner Ex Message: The EXECUTE permission was denied on the object 'gs_GetVersion', database 'GSPDB', schema 'dbo'.

Inner Ex Source: .Net SqlClient Data Provider

Inner Ex Target Site: Void OnError(System.Data.SqlClient.SqlException, Boolean)

Inner Ex Stack Trace

```
at System.Data.SqlClient.SqlConnection.OnError(SqlException exception, Boolean breakConnection)
at System.Data.SqlClient.SqlInternalConnection.OnError(SqlException exception, Boolean breakConnection)
at System.Data.SqlClient.TdsParser.ThrowExceptionAndWarning(TdsParserStateObject stateObj)
at System.Data.SqlClient.TdsParser.Run(RunBehavior runBehavior, SqlCommand cmdHandler, SqlDataReader dataStream, BulkCopySimpleResultSet bulkCopyHandler, TdsParserStateObject stateObj)
at System.Data.SqlClient.SqlDataReader.ConsumeMetaData()
at System.Data.SqlClient.SqlDataReader.get_MetaData()
at System.Data.SqlClient.SqlCommand.FinishExecuteReader(SqlDataReader ds, RunBehavior runBehavior, String resetOptionsString)
at System.Data.SqlClient.SqlCommand.RunExecuteReaderTds(CommandBehavior cmdBehavior, RunBehavior runBehavior, Boolean returnStream, Boolean async)
at System.Data.SqlClient.SqlCommand.RunExecuteReader(CommandBehavior cmdBehavior, RunBehavior runBehavior, Boolean returnStream, String method, DbAsyncResult result)
at System.Data.SqlClient.SqlCommand.RunExecuteReader(CommandBehavior cmdBehavior, RunBehavior runBehavior, Boolean returnStream, String method)
at System.Data.SqlClient.SqlCommand.ExecuteScalar()
at GalleryServerPro.Data.SqlServer.SqlDataProvider.GetDataSchemaVersionString()
```

Defect Details

```
at GalleryServerPro.Data.SqlServer.SqlDataProvider.VerifySchemaVersion()
at GalleryServerPro.Data.SqlServer.SqlDataProvider.InitializeDataStore()
at GalleryServerPro.Business.AppSetting.Initialize(ApplicationTrustLevel trustLevel, String physicalAppPath,
String appName)
at GalleryServerPro.Web.Util.InitializeBusinessLayer()
at GalleryServerPro.Web.Util.InitializeApplication()
at GalleryServerPro.Web.Pages.GalleryPage..cctor()
```

This error occurs only when using a SQL Server login account that has the minimum number of permissions required to run Gallery Server. This type of account is created when a user specifies a new SQL Server account for the runtime environment during installation.

Resolution

The SQL script that upgrades the database from data schema version 2.1.3162 to 2.3.3421 did not re-apply EXECUTE permission to the user-defined function gs_GetVersion after it is recreated with the latest version number. The permission can be applied by executing the following SQL:

```
GRANT EXECUTE ON [dbo].[gs_GetVersion] TO [gs_GalleryServerProRole]
```

The fix is to update the upgrade script to include this statement after recreating the function. The following files were updated:

```
TIS.GSP.Data.SqlServer\SqlUpgrade_2_1_3162_to_2_3_3421_SqlServer2000.sql
TIS.GSP.Data.SqlServer\SqlUpgrade_2_1_3162_to_2_3_3421_SqlServer2005.sql
```

Defect Id: 227

Project: Gallery Server Pro

Name: Restricted permission user may not be able to download objects in ZIP archive

Date Found: 7/2/2009

Build #: 2.3.3456

Severity: Medium Impact

Date Fixed: 7/2/2009

Build # of Fix: 2.3.3470

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: Medium

Actual Work: 1 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

When a user is restricted from viewing certain albums, Gallery Server may show the viewable albums in a virtual container that does not represent an actual album. In this circumstance, if the user clicks the Download objects menu item, the following error will appear:

```
User 'XXX' does not have permission 'ViewAlbumOrMediaObject' for album ID -2147483648.
```

Resolution

Two fixes were required:

1. The page load event checked to ensure the user has view permissions to the current album by executing this function:

```
CheckUserSecurity(SecurityActions.ViewAlbumOrMediaObject);
```

Defect Details

However, this is not necessary as the GalleryPage.GetAlbum method automatically retrieves only those objects the user has permission to view, so it was removed.

2. The previous fix resolved the immediate error message, but a new error occurred if the user chose to download an album contained within the virtual album. The code that creates the ZIP archive has this signature:

public

StreamCreateZipStream(intparentAlbumId,List<int> albumIds,List<int> mediaObjectIds,DisplayObjectTypeimageSize)When the parent album is virtual, the parentAlbumId parameter is Int32.MinValue, which caused an InvalidAlbumException later in the method. The fix was to modify this method by catching the exception and handling this special situation in the section that iterates through each of the album ID's and media object ID's.

Defect Id: 228

Project: Gallery Server Pro

Name: Image metadata sometimes exceeds window's borders

Date Found: 7/28/2009

Build #: 2.3.3456

Severity: Medium Impact

Date Fixed: 7/28/2009

Build # of Fix: 2.3.3492

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: Medium

Actual Work: 1 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

When there is more metadata than can fit in the popup window, it can run outside the border. Scrollbars appear if the data goes about 20px past the border, but the scrollbars should appear right at the border, not below it.

Resolution

Edited the CSS class that defines the popup window's content. It is in ca_styles.css and was this:

```
.gsp_ns
```

```
.dg2ContentCss {min-height:100px;height:100%;max-height:1460px;overflow:auto;}
```

It was changed to this:

```
.gsp_ns
```

```
.dg2ContentCss {height:460px;overflow:auto;}
```

This change caused scrollbars to appear on the bottom and right sides where there is a lot of metadata. This causes an aesthetic problem because the scrollbars have square corners and the bottom of the metadata popup has round corners. To fix this, the styling was changed so that the top corners retained the rounded effect but the bottom had square corners. To do this, first I added a new CSS class to gallery.css:

```
.gsp_roundedtop10
```

```
{-moz-border-radius:10px10px00;-webkit-border-top-right-radius:10px;-webkit-border-top-left-radius:10px;}
```

Then I specified this class instead of gsp_rounded10 in the definition of the Dialog control in gs\controls\mediaobjectview.ascx.

Defect Details

Defect Id: 229

Project: Gallery Server Pro

Name: Expand/collapse image button not working on Add objects page

Date Found: 7/30/2009

Build #: 2.3.3456

Severity: Low Impact

Date Fixed: 7/30/2009

Build # of Fix: 2.3.3498

Status: Closed

Estimated Work: 30 min

Assigned To: Roger Martin

Priority: Low

Actual Work: 15 min

Reported By: Roger Martin

Percent Complete: 100

Description

The Add objects page has an expand/collapse image button that shows or hides a few options when uploading files. This image should toggle between two images when clicked, but this is not happening.

Resolution

Added the following code to the function ConfigureControlsFirstTime in addobjects.ascx.cs:

```
cpe.ExpandedImage =  
Util.GetUrl("/images/collapse.jpg");  
cpe.CollapsedImage =Util.GetUrl("/images/expand.jpg");  
imgExpCol.ImageUrl =Util.GetUrl("/images/expand.jpg");
```

Defect Id: 230

Project: Gallery Server Pro

Name: Current tab in site admin menu not shown as selected when URL rewriting is used

Date Found: 8/3/2009

Build #: 2.3.3456

Severity: Medium Impact

Date Fixed: 8/3/2009

Build # of Fix: 2.3.3502

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: Low

Actual Work: 6 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

In a normal GSP installation, the current tab in the Site admin menu is shown as sunken and the font is shown in a dark red color. When GSP is placed in a web site where URL rewriting is used, the current tab does not have this effect. Instead, it appears the same as the other tabs.

Resolution

Ultimately, this is caused by an issue inside the base class used in the ComponentArt NavBar control. Specifically, there is logic that determines whether the current tab should be highlighted based on whether its URL matches the NavigationUrl. When rewriting is used, these do not match.

The fix was to add an additional check to the function UpdateBestMatch() in BaseNavigator.cs:

```
pathsEqual = (pathsEqual || Context.Request.RawUrl.ToLower() == givenUrl.PathAndQuery.ToLower());
```

This line is placed just after the line where the pathsEqual variable is defined.

Defect Details

Defect Id: 231

Project: Gallery Server Pro

Name: (SQLite) Cannot use encrypted passwords

Date Found: 8/12/2009

Build #: 2.3.3510

Severity: Low Impact

Date Fixed: 8/12/2009

Build # of Fix: 2.3.3511

Status: Closed

Estimated Work: 1 hrs

Assigned To: Roger Martin

Priority: Medium

Actual Work: 1 hrs

Reported By: Roger Martin

Percent Complete: 100

Description

The GetPassword function in the SQLite membership provider does not correctly decode encrypted passwords. A default installation of GSP stores passwords in clear text, so this bug only affects users who change their membership settings to use encrypted passwords.

Resolution

The GetPassword function in SQLiteMembershipProvider calls the UnEncodePassword function, and this is where the bug lies. This was the original function:

```
private string UnEncodePassword(string encodedPassword, MembershipPasswordFormat passwordFormat)
{
    // Decrypts or leaves the password clear based on the PasswordFormat.
    string password = encodedPassword;

    switch (passwordFormat)
    {
        case MembershipPasswordFormat.Clear:
            break;
        case MembershipPasswordFormat.Encrypted:
            password = Encoding.Unicode.GetString(DecryptPassword(Convert.FromBase64String(password)));
            break;
        case MembershipPasswordFormat.Hashed:
            throw new ProviderException("Cannot unencode a hashed password.");
        default:
            throw new ProviderException("Unsupported password format.");
    }

    return password;
}
```

Here is the fixed function. Notice the first several characters of the decrypted password are removed.

```
private string UnEncodePassword(string encodedPassword, MembershipPasswordFormat passwordFormat)
{
    // Decrypts or leaves the password clear based on the PasswordFormat.
    string password = encodedPassword;

    switch (passwordFormat)
    {
        case MembershipPasswordFormat.Clear:
            break;
```

Defect Details

```
case MembershipPasswordFormat.Encrypted:
    byte[] bytes = base.DecryptPassword(Convert.FromBase64String(password));
    if (bytes == null)
    {
        password = null;
    }
    else
    {
        password = Encoding.Unicode.GetString(bytes, 0x10, bytes.Length - 0x10);
    }
    break;
case MembershipPasswordFormat.Hashed:
    throw new ProviderException("Cannot unencode a hashed password.");
default:
    throw new ProviderException("Unsupported password format.");
}

return password;
}
```