

MARKETS GATEWAY BROWSERLESS

C# .NET VIA CERT AUTHENTICATION

INSTRUCTIONS & CODING EXAMPLES

8/11/2023

CONFIGURING C# FOR MARKETS GATEWAY SANDBOX

A C# project was developed for the purposes of connecting to the Markets Gateway Sandbox which utilizes Token Authentication, and is able to send / receive XML.

- #1 Open Visual Studio. **Note:** Code was last confirmed to work using Visual Studio Professional 2022.
- #2 Copy the code provided below into your project.
- #3 Invoke the various methods of the class below (provided later) to accomplish the sending, and receiving of XML to Markets Gateway.
- #4 In the code below: Provide your own certificate name and certificate password for the line containing `-> certificates.Import("yourCertName", "yourCertPassword",...`

(WebControl.cs):

C# Markets Gateway Class

```
using System;
using System.Linq;
using System.Collections.Generic;
using System.Text;
using System.Net;
using System.IO;
using System.Windows.Forms;
using System.ComponentModel;
using System.Globalization;
using System.Security.Cryptography.X509Certificates;

namespace MarketsGatewayUsingCerts
{
    static class Program
    {
        /// <summary>
        /// The main entry point for the application.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new frmGateway());
        }
    }
}

namespace MarketsGatewayUsingCerts
{
    class WebControl
    {
        public string User { get; set; }
        public string Password { get; set; }
        public string XML_SendStr { get; set; }
        public string OpenAM_URL { get; set; }
        public string Web_URL { get; set; }
        private string Token { get; set; }
        public enum XMLTypes { XMLQuery, XMLSubmit };
        public XMLTypes XMLType { get; set; }
        private const string querySoapAction = "/xml/query";
    }
}
```

```

private const string submitSoapAction = "/xml/submit";
public string sendXMLRoutine()
{
    string soapaction;
    HttpPost hp = new HttpPost();
    string response = "";
    soapaction = (XMLType == XMLTypes.XMLQuery) ? querySoapAction : submitSoapAction;
    hp.Password = Password;
    hp.User = User;
    hp.WebURL = OpenAM_URL;
    Token = hp.openamGetToken();
    hp = null;
    hp = new HttpPost();
    hp.TokenID = Token;
    hp.WebURL = Web_URL + soapaction;

    if (hp.systemConnect())
    {
        //Send your packet
        response = hp.Upload(XML_SendStr);
        hp = null;
    }
    return response;
}
}
class HttpPost
{
    private HttpWebRequest Connection;
    private Dictionary<string, string> openAM_KeyValuePairs;
    private string url;
    private bool isOpenAM = false;
    public string User { get; set; }
    public string Password { get; set; }
    public string TokenID { get; set; }
    public string WebURL
    {
        get { return url; }
        set
        {
            url = value;
            if (Connection == null)
            {
                try
                {
                    Connection = (HttpWebRequest)WebRequest.Create(url);
                }
                catch (Exception e)
                {
                    MessageBox.Show("Exception attempting to connect:" + e.Message);
                }
            }
        }
    }
}
public string openamGetToken()
{
    string tokenResult = "";
    StringBuilder responseData = null;
    try

```

```

    {
        if (User != "" && Password != "")
        {
            X509Certificate2Collection certificates = new X509Certificate2Collection();
            certificates.Import("yourCertName", "yourCertPassword",
            X509KeyStorageFlags.MachineKeySet | X509KeyStorageFlags.PersistKeySet);
            Connection.ClientCertificates = certificates;

            Connection.ContentType = "application/json";
            Connection.Headers.Set("X-OpenAM-Username", User);
            Connection.Headers.Set("X-OpenAM-Password", Password);
            Connection.ContentLength = 0;
            Connection.ProtocolVersion = System.Net.HttpVersion.Version11;
            Connection.Method = "POST";
            Connection.Accept = "*/*";
            Connection.KeepAlive = true;

            //Get the response back and store into respondedata
            responseData = GetResponseData();
            openAM_KeyValuePairs = responseData.ToString().Remove(0, 1).Split(',').Select(value =>
value.Split(':'))
            .ToDictionary(pair => pair[0].Replace("\\"", "\""), pair => pair[1].Replace("\\"", "\""));
            //Get the tokenid and use as Cookie back in main request
            tokenResult = openAM_KeyValuePairs["tokenId"];
        }
        else
        {
            MessageBox.Show("You must first provide a username and password");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show("Exception attempting to obtain token:" + ex.Message);
    }
    finally
    {
        //Connection.KeepAlive = false;
        Connection = null;
    }
    return tokenResult;
}
//Call this After openAM_GetToken Routine
public bool systemConnect()
{
    bool connectionConfigured = false;
    try
    {
        if (TokenID != "")
        {
            Connection.ContentType = "text/xml";
            Connection.ProtocolVersion = System.Net.HttpVersion.Version11;
            Connection.Method = "POST";
            Connection.KeepAlive = false;
            Connection.Headers.Set("Cookie", "pjmauthtrain="+ TokenID);
            //Uncomment this line if using SUMA but provide sub-account
            //Connection.Headers.Set("XParticipantName", sub-account);
            connectionConfigured = true;
        }
    }
}

```

```

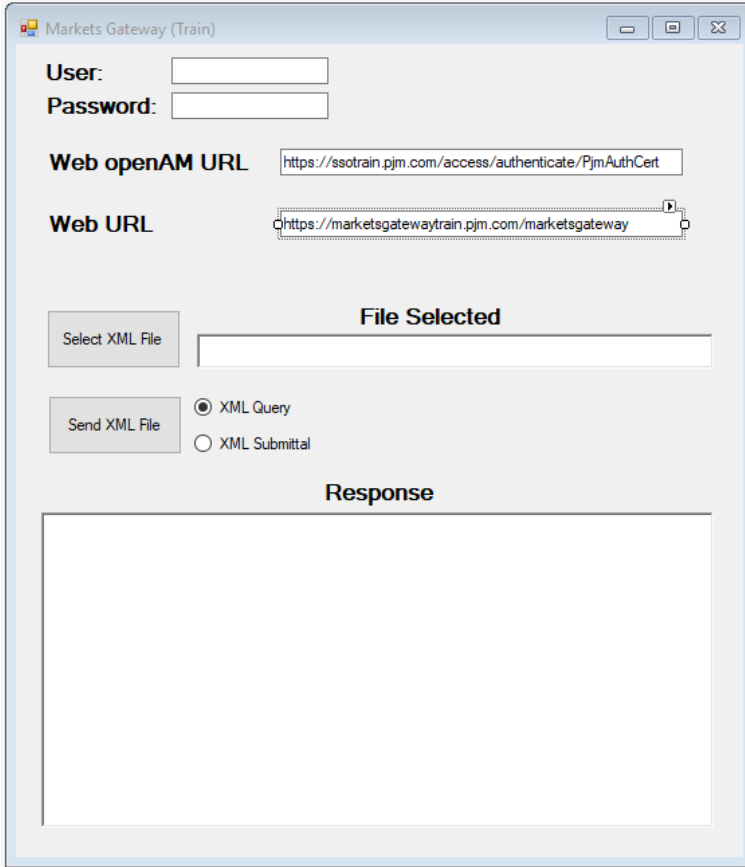
        else
        {
            MessageBox.Show("No token ID provided");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show("Exception attempting to connect to web using token:" + ex.Message);
    }
    return connectionConfigured;
}
public string Upload(string uploadString)
{
    StringBuilder rd = null;
    byte[] byteArray = Encoding.ASCII.GetBytes(uploadString);
    Connection.ContentLength = byteArray.Length;
    Stream postStream = Connection.GetRequestStream();
    postStream = Connection.GetRequestStream();
    postStream.Write(byteArray, 0, byteArray.Length);
    postStream.Close();
    //Assign the response object of 'WebRequest' to a 'WebResponse' variable.
    rd = GetResponseData();
    Connection = null;
    return rd.ToString();
}
private StringBuilder GetResponseData()
{
    StringBuilder rd = new StringBuilder();
    WebResponse WebResponseObject = Connection.GetResponse();
    using (Stream stream = WebResponseObject.GetResponseStream())
    {
        StreamReader reader = new StreamReader(stream, Encoding.UTF8);
        rd.Append(reader.ReadToEnd());
    }
    WebResponseObject.Close();
    WebResponseObject = null;
    return rd;
}
}
}

```

C# Sample code for the Sandbox that utilizes the class above

A form was created to utilize, and to invoke the appropriate routines defined in the class code above. A screenshot of the form is provided below.

NOTE: The code above is exclusively for the sandbox due to the cookie definition highlighted in **Yellow** above. You could easily make that a passed in parameter to generalize for the sandbox (pjmauthtrain) / production (pjmauth) environment. Additionally, for the form below if you remove the references to "train" then it will be configured to work for the Production environment.



#2 The code that creates the above screen is as follows (MarketsGatewayForm.designer.cs):

```
namespace MarketsGatewayUsingCerts
{
    partial class frmGateway
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed; otherwise, false.</param>
```

```

protected override void Dispose(bool disposing)
{
    if (disposing && (components != null))
    {
        components.Dispose();
    }
    base.Dispose(disposing);
}

#region Windows Form Designer generated code
/// <summary>
/// Required method for Designer support - do not modify
/// the contents of this method with the code editor.
/// </summary>
private void InitializeComponent()
{
    this.txtResponse = new System.Windows.Forms.RichTextBox();
    this.lblResults = new System.Windows.Forms.Label();
    this.btnSelectFile = new System.Windows.Forms.Button();
    this.btnUploadFile = new System.Windows.Forms.Button();
    this.lblFileSelected = new System.Windows.Forms.Label();
    this.lblUser = new System.Windows.Forms.Label();
    this.lblPassword = new System.Windows.Forms.Label();
    this.txtUser = new System.Windows.Forms.TextBox();
    this.txtPassword = new System.Windows.Forms.TextBox();
    this.txtOpenAMURL = new System.Windows.Forms.TextBox();
    this.lblURL = new System.Windows.Forms.Label();
    this.txtGatewayURL = new System.Windows.Forms.TextBox();
    this.lblWebURL = new System.Windows.Forms.Label();
    this.radQuery = new System.Windows.Forms.RadioButton();
    this.radSubmit = new System.Windows.Forms.RadioButton();
}
    
```

```

this.txtFileSelected = new System.Windows.Forms.RichTextBox();

this.SuspendLayout();

//

// txtResponse

//

this.txtResponse.DetectUrls = false;

this.txtResponse.Location = new System.Drawing.Point(19, 349);

this.txtResponse.Name = "txtResponse";

this.txtResponse.Size = new System.Drawing.Size(501, 235);

this.txtResponse.TabIndex = 0;

this.txtResponse.Text = "";

//

// lblResults

//

this.lblResults.AutoSize = true;

this.lblResults.Font = new System.Drawing.Font("Microsoft Sans Serif", 12F, System.Drawing.FontStyle.Bold,
System.Drawing.GraphicsUnit.Point, ((byte)0));

this.lblResults.Location = new System.Drawing.Point(226, 324);

this.lblResults.Name = "lblResults";

this.lblResults.Size = new System.Drawing.Size(90, 20);

this.lblResults.TabIndex = 1;

this.lblResults.Text = "Response";

//

// btnSelectFile

//

this.btnSelectFile.Location = new System.Drawing.Point(23, 198);

this.btnSelectFile.Name = "btnSelectFile";

this.btnSelectFile.Size = new System.Drawing.Size(100, 44);

this.btnSelectFile.TabIndex = 2;

this.btnSelectFile.Text = "Select XML File";
    
```



```

this.btnSelectFile.UseVisualStyleBackColor = true;

this.btnSelectFile.Click += new System.EventHandler(this.btnFileSelected_Click);

//

// btnUploadFile

//

this.btnUploadFile.Location = new System.Drawing.Point(24, 262);

this.btnUploadFile.Name = "btnUploadFile";

this.btnUploadFile.Size = new System.Drawing.Size(100, 44);

this.btnUploadFile.TabIndex = 3;

this.btnUploadFile.Text = "Send XML File";

this.btnUploadFile.UseVisualStyleBackColor = true;

this.btnUploadFile.Click += new System.EventHandler(this.btnSendFile_Click);

//

// lblFileSelected

//

this.lblFileSelected.AutoSize = true;

this.lblFileSelected.Font = new System.Drawing.Font("Microsoft Sans Serif", 12F, System.Drawing.FontStyle.Bold,
System.Drawing.GraphicsUnit.Point, ((byte)0));

this.lblFileSelected.Location = new System.Drawing.Point(252, 193);

this.lblFileSelected.Name = "lblFileSelected";

this.lblFileSelected.Size = new System.Drawing.Size(114, 20);

this.lblFileSelected.TabIndex = 4;

this.lblFileSelected.Text = "File Selected";

//

// lblUser

//

this.lblUser.AutoSize = true;

this.lblUser.Font = new System.Drawing.Font("Microsoft Sans Serif", 12F, System.Drawing.FontStyle.Bold,
System.Drawing.GraphicsUnit.Point, ((byte)0));

this.lblUser.Location = new System.Drawing.Point(18, 11);

this.lblUser.Name = "lblUser";
    
```

```

this.lblUser.Size = new System.Drawing.Size(52, 20);

this.lblUser.TabIndex = 6;

this.lblUser.Text = "User:";

//

// lblPassword

//

this.lblPassword.AutoSize = true;

this.lblPassword.Font = new System.Drawing.Font("Microsoft Sans Serif", 12F, System.Drawing.FontStyle.Bold,
System.Drawing.GraphicsUnit.Point, ((byte)0));

this.lblPassword.Location = new System.Drawing.Point(19, 36);

this.lblPassword.Name = "lblPassword";

this.lblPassword.Size = new System.Drawing.Size(91, 20);

this.lblPassword.TabIndex = 7;

this.lblPassword.Text = "Password:";

//

// txtUser

//

this.txtUser.Location = new System.Drawing.Point(116, 10);

this.txtUser.Name = "txtUser";

this.txtUser.Size = new System.Drawing.Size(117, 20);

this.txtUser.TabIndex = 8;

//

// txtPassword

//

this.txtPassword.Location = new System.Drawing.Point(116, 36);

this.txtPassword.Name = "txtPassword";

this.txtPassword.PasswordChar = '*';

this.txtPassword.Size = new System.Drawing.Size(117, 20);

this.txtPassword.TabIndex = 9;

//
    
```

```

// txtOpenAMURL
//
this.txtOpenAMURL.Location = new System.Drawing.Point(197, 78);
this.txtOpenAMURL.Name = "txtOpenAMURL";
this.txtOpenAMURL.Size = new System.Drawing.Size(300, 20);
this.txtOpenAMURL.TabIndex = 11;
this.txtOpenAMURL.Text = "https://ssotrain.pjm.com/access/authenticate/PjmAuthCert";
//
// lblURL
//
this.lblURL.AutoSize = true;
this.lblURL.Font = new System.Drawing.Font("Microsoft Sans Serif", 12F, System.Drawing.FontStyle.Bold,
System.Drawing.GraphicsUnit.Point, ((byte)0));
this.lblURL.Location = new System.Drawing.Point(21, 78);
this.lblURL.Name = "lblURL";
this.lblURL.Size = new System.Drawing.Size(157, 20);
this.lblURL.TabIndex = 10;
this.lblURL.Text = "Web openAM URL";
//
// txtGatewayURL
//
this.txtGatewayURL.Location = new System.Drawing.Point(197, 124);
this.txtGatewayURL.Name = "txtGatewayURL";
this.txtGatewayURL.Size = new System.Drawing.Size(300, 20);
this.txtGatewayURL.TabIndex = 13;
this.txtGatewayURL.Text = "https://marketsgatewaytrain.pjm.com/marketsgateway";
//
// lblWebURL
//
this.lblWebURL.AutoSize = true;

```

```

this.lblWebURL.Font = new System.Drawing.Font("Microsoft Sans Serif", 12F, System.Drawing.FontStyle.Bold,
System.Drawing.GraphicsUnit.Point, ((byte)0));

this.lblWebURL.Location = new System.Drawing.Point(21, 124);

this.lblWebURL.Name = "lblWebURL";

this.lblWebURL.Size = new System.Drawing.Size(86, 20);

this.lblWebURL.TabIndex = 12;

this.lblWebURL.Text = "Web URL";

//

// radQuery

//

this.radQuery.AutoSize = true;

this.radQuery.Checked = true;

this.radQuery.Location = new System.Drawing.Point(133, 262);

this.radQuery.Name = "radQuery";

this.radQuery.Size = new System.Drawing.Size(78, 17);

this.radQuery.TabIndex = 14;

this.radQuery.TabStop = true;

this.radQuery.Text = "XML Query";

this.radQuery.UseVisualStyleBackColor = true;

//

// radSubmit

//

this.radSubmit.AutoSize = true;

this.radSubmit.Location = new System.Drawing.Point(133, 289);

this.radSubmit.Name = "radSubmit";

this.radSubmit.Size = new System.Drawing.Size(93, 17);

this.radSubmit.TabIndex = 15;

this.radSubmit.Text = "XML Submittal";

this.radSubmit.UseVisualStyleBackColor = true;

//
    
```

```

// txtFileSelected
//
this.txtFileSelected.DetectUrls = false;
this.txtFileSelected.Location = new System.Drawing.Point(135, 216);
this.txtFileSelected.Multiline = false;
this.txtFileSelected.Name = "txtFileSelected";
this.txtFileSelected.ScrollBars = System.Windows.Forms.RichTextBoxScrollBars.Horizontal;
this.txtFileSelected.Size = new System.Drawing.Size(385, 26);
this.txtFileSelected.TabIndex = 16;
this.txtFileSelected.Text = "";
this.txtFileSelected.WordWrap = false;
//
// frmGateway
//
this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
this.ClientSize = new System.Drawing.Size(542, 606);
this.Controls.Add(this.txtFileSelected);
this.Controls.Add(this.radSubmit);
this.Controls.Add(this.radQuery);
this.Controls.Add(this.txtGatewayURL);
this.Controls.Add(this.lblWebURL);
this.Controls.Add(this.txtOpenAMURL);
this.Controls.Add(this.lblURL);
this.Controls.Add(this.txtPassword);
this.Controls.Add(this.txtUser);
this.Controls.Add(this.lblPassword);
this.Controls.Add(this.lblUser);
this.Controls.Add(this.lblFileSelected);
this.Controls.Add(this.btnUploadFile);
    
```

```

this.Controls.Add(this.btnSelectFile);

this.Controls.Add(this.lblResults);

this.Controls.Add(this.txtResponse);

this.Name = "frmGateway";

this.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen;

this.Text = "Markets Gateway (Train)";

this.ResumeLayout(false);

this.PerformLayout();

}

```

```
#endregion
```

```

private System.Windows.Forms.RichTextBox txtResponse;

private System.Windows.Forms.Label lblResults;

private System.Windows.Forms.Button btnSelectFile;

private System.Windows.Forms.Button btnUploadFile;

private System.Windows.Forms.Label lblFileSelected;

private System.Windows.Forms.Label lblUser;

private System.Windows.Forms.Label lblPassword;

private System.Windows.Forms.TextBox txtUser;

private System.Windows.Forms.TextBox txtPassword;

private System.Windows.Forms.TextBox txtOpenAMURL;

private System.Windows.Forms.Label lblURL;

private System.Windows.Forms.TextBox txtGatewayURL;

private System.Windows.Forms.Label lblWebURL;

private System.Windows.Forms.RadioButton radQuery;

private System.Windows.Forms.RadioButton radSubmit;

private System.Windows.Forms.RichTextBox txtFileSelected;

}

}

```

#3 The code that resides behind the screen above, and drives the screen, which also defines the code behind the two critical buttons is provided below.

(MarketsGatewayForm.cs) :

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace MarketsGatewayUsingCerts
{
    public partial class frmGateway : Form
    {
        private string XML_to_Upload;
        public frmGateway()
        {
            InitializeComponent();
        }
        private void btnFileSelected_Click(object sender, EventArgs e)
        {
            // Create an instance of the open file dialog box.
            OpenFileDialog openFileDialog1 = new OpenFileDialog();
            // Set filter options and filter index.
            openFileDialog1.Filter = "XML Files (*.xml)|*.xml|Text Files (.txt)|*.txt";
            openFileDialog1.FilterIndex = 1;
            openFileDialog1.Multiselect = false;
            // Call the ShowDialog method to show the dialog box.
            DialogResult userClickedOK = openFileDialog1.ShowDialog();
            // Process input if the user clicked OK.
            if (userClickedOK == DialogResult.OK)
            {
                this.txtFileSelected.Text = openFileDialog1.FileName;
                System.IO.StreamReader sr = new System.IO.StreamReader(this.txtFileSelected.Text);
                XML_to_Upload = sr.ReadToEnd();
                sr.Close();
            }
        }
        private void btnSendFile_Click(object sender, EventArgs e)
        {
            if (txtUser.Text != "" && txtPassword.Text != "")
            {
                WebControl c = new WebControl();
                c.User = txtUser.Text;
                c.Password = txtPassword.Text;
                c.XML_SendStr = XML_to_Upload;
                c.OpenAM_URL = this.txtOpenAMURL.Text;
                c.Web_URL = this.txtGatewayURL.Text;
                c.XMLType = (this.radQuery.Checked) ? WebControl.XMLTypes.XMLQuery :
                WebControl.XMLTypes.XMLSubmit;
                this.txtResponse.Text = "Sending...";
                this.txtResponse.Refresh();
                this.txtResponse.Text = c.sendXMLRoutine();
            }
        }
    }
}

```

```
}  
  }  
} }
```