



## Coding Examples for Browserless HTTP Request with Token Based Authentication

The following are example code snippets for use with the MSRS browserless HTTP Request with Token Based Authentication and PKI certificate usage to download MSRS reports.

For more details regarding PKI certificates please refer to PJM's [PKI Authentication Guide](#).

To access our Train(sandbox) environment, the URLs in each example need to be replaced with the Train URLs below.

### **Train(sandbox):**

Authentication URL: <https://ssotrain.pjm.com/access/authenticate/pjmauthcert>

MSRS URL: <https://msrsapptrain.pjm.com>

Authentication header cookie: pjmauthtrain =

### **Production:**

Authentication URL: <https://sso.pjm.com/access/authenticate/pjmauthcert>

MSRS URL: <https://msrsapp.pjm.com>

Authentication header cookie: pjmauth =

To see the list of available reports to download, authentication must first be made and then a request can be sent to the following URL:

The following URL will generate a list of available reports in XML format.

<https://msrsapp.pjm.com/msrs/rest/secure/download/reportNames>



## Coding Examples for Browserless HTTP Request with Token Based Authentication

### curl

It is standard as part of both linux and Windows 10. It can also be downloaded at: <https://curl.haxx.se/download.html>.

#### First authenticate with credentials to receive token:

```
curl --request POST -key testcert.key -cert 'testcert.crt:<privatekwyppassword>' --header "X-OpenAM-Username: username" --header "X-OpenAM-password: password" --header "Content-Type:application/json" --data "{}" https://sso.pjm.com/access/authenticate/pjmauthcert
```

#### Response returns token:

```
{"tokenId":"9-rN8J5nYazraGSKA6DRY0B80C4.*AAJTSQACMDIAA1NLABxQzm1tdGRUK3R0RUFWR01Ic3dDMTVtWHAzTjg9AAJTMQACMDE.*", "successUrl":"/access/console", "realm":"/"}
```

#### Use received Token to request MSRS download:

```
curl --request GET --header "Cookie: pjmauth=9-rN8J5nYazraGSKA6DRY0B80C4.*AAJTSQACMDIAA1NLABxQzm1tdGRUK3R0RUFWR01Ic3dDMTVtWHAzTjg9AAJTMQACMDE.*" --header "Content-Type:text/csv" "https://msrsapp.pjm.com/msrs/rest/secure/download/reports?version=L&shortName=weeklybillingstatement-csvandxml&stop=01-07-2020&format=C&start=01-01-2020"
```

#### Use received Token to request MSRS download (SUMA users to access other account):

```
curl --request GET --header "Cookie: pjmauth=9-rN8J5nYazraGSKA6DRY0B80C4.*AAJTSQACMDIAA1NLABxQzm1tdGRUK3R0RUFWR01Ic3dDMTVtWHAzTjg9AAJTMQACMDE.*" --header "Content-Type:text/csv" -header "Suma-Account: ABCD" "https://msrsapp.pjm.com/msrs/rest/secure/download/reports?version=L&shortName=weeklybillingstatement-csvandxml&stop=01-07-2020&format=C&start=01-01-2020"
```

#### Use received Token to request MSRS download (SUMA users to request for all accounts):

```
curl --request GET --header "Cookie: pjmauth=9-rN8J5nYazraGSKA6DRY0B80C4.*AAJTSQACMDIAA1NLABxQzm1tdGRUK3R0RUFWR01Ic3dDMTVtWHAzTjg9AAJTMQACMDE.*" --header "Content-Type:text/csv" "https://msrsapp.pjm.com/msrs/rest/secure/download/reports?version=L&shortName=weeklybillingstatement-csvandxml&stop=01-07-2020&format=C&start=01-01-2020&requestAll=true"
```



## Coding Examples for Browserless HTTP Request with Token Based Authentication

### Java (Java version 7+ and CLI is required)

```
/**
 * Example call to download a file from PJM into a FileOutputStream to save
 * the results to disk.
 */
public static void downloadToFile() {
    try {
        final PjmRemoteCommand command = new PjmRemoteCommand();
        // first set up URL and credentials
        command.setServiceUrl("https://msrsapp.pjm.com/msrs/");
        command.setUsername("pjmuser");
        command.setPassword("Pjm4567!");
        command.setCertificate(new File(<<path to .pfx or .p12 file>>));
        command.setCertificatePassword(<<certificate_password>>);

        // optional header parameter for SUMA users to access other account
        // command.getHeaderParams().put("Suma-Account", "ABCD");

        // the URL action you want to execute
        command.setAction("rest/secure/download/reports");

        // any query parameters required by the download
        command.getQueryParams().put("shortName", Arrays.asList("weeklybillingstatement-csvandxml"));
        command.getQueryParams().put("start", Arrays.asList("01-01-2020"));
        command.getQueryParams().put("stop", Arrays.asList("01-07-2020"));
        command.getQueryParams().put("format", Arrays.asList("C"));
        command.getQueryParams().put("version", Arrays.asList("L"));

        // optional query parameter for SUMA users to request for all accounts
        // command.getQueryParams().put("requestAll", Arrays.asList("true"));

        // create an output stream to capture the output result
    }
}
```



## Coding Examples for Browserless HTTP Request with Token Based Authentication

```
final File file = new File("weeklybillingstatement-csvandxml.csv");
final FileOutputStream fos = new FileOutputStream(file);

// buffer the stream for performance
final BufferedOutputStream bos = new BufferedOutputStream(fos);
command.setOutputStream(bos);

// execute the command and inspect the results
final PjmResult result = PjmClient.execute(command);

if (result.getResultCode() == PjmResult.CLI_SUCCESS) {
    System.out.println("Download succeeded for file: " + result.getFileName());
} else {
    System.out.println("Download failed for file: " + result.getFileName());
}
} catch (final Exception ex) {
    ex.printStackTrace();
}
}
```



## Coding Examples for Browserless HTTP Request with Token Based Authentication

### .Net (Java version 7+ and CLI is required)

```
static void Main(string[] args)
{
    // Set up URL and credentials
    string serviceUrl = "https://msrsapp.pjm.com/msrs/";
    string userName = "pjmuser";
    string password = "Pjm4567!";
    string certPath = "<<path to .pfx or .p12 file>>";
    string certPassword = "<<certPassword>>";

    // the URL action to excute
    string action = "rest/secure/download/reports";

    // Set up the report parameters
    string shortName = "weeklybillingstatement-csvandxml";
    string start = "01-01-2020";
    string stop = "01-07-2020";
    string format = "C";
    string version = "L";

    //Create process
    System.Diagnostics.Process pProcess = new System.Diagnostics.Process();

    //strCommand is path and file name of command to run
    pProcess.StartInfo.FileName = "java";

    //Optional
    pProcess.StartInfo.WorkingDirectory = "C:/Personal/tools/pjm-cli/";

    //strCommandParameters are parameters to pass to program
    string argParams = "-Xms64m -Xmx256m -jar pjm-cli.jar -u " + userName + " -p " + password
```



## Coding Examples for Browserless HTTP Request with Token Based Authentication

```
+ " -d ./download/-z 180000 -r\" + certPath + "|" + certPassword + "\" -s " + serviceUrl + " -a " + action + " -q  
shortName=" + shortName + " -q start=" + start + " -q stop=" + stop + " -q version=" + version  
+ " -q format=" + format;  
  
//Optional param for SUMA users to request for all accounts  
//argParams = argParams + " -q requestAll=true";  
  
//Optional param for SUMA users to access other account  
//argParams = argParams + " -h Suma-Account=ABCD";  
  
pProcess.StartInfo.Arguments = argParams;  
  
// hide window  
pProcess.StartInfo.CreateNoWindow = true;  
pProcess.StartInfo.UseShellExecute = false;  
  
//Set output of program to be written to process output stream  
pProcess.StartInfo.RedirectStandardOutput = true;  
  
//Start the process  
pProcess.Start();  
  
//Get program output  
string strOutput = pProcess.StandardOutput.ReadToEnd();  
  
//Wait for process to finish  
pProcess.WaitForExit();  
  
Console.WriteLine(strOutput);  
Console.ReadLine();  
}
```



## Coding Examples for Browserless HTTP Request with Token Based Authentication

First install certificate and extract Public Key from browser. Steps can be found in [PKI Exporting Public Keys](#) document in example 2.

```
Dim Request As Object
Dim url As String
```

```
Set Request = CreateObject("WinHttp.WinHttpRequest.5.1")

url = "https://sso.pjm.com/access/authenticate/pjmauthcert"

Request.Open "POST", url, False
Request.SetRequestHeader "X-OpenAM-Username", username
Request.SetRequestHeader "X-OpenAM-Password", password
Request.SetClientCertificate "certificate CN"
Request.Send
```

Where username = PJM user account username  
password = PJM user account password  
certificate CN = CN from user's PKI certificate uploaded in Account Manager

Response will return authentication Token which can be stored/saved

```
{ "tokenId": "AQIC5w_TEXT_REMOVED_NTcy*", "successUrl": "/openam/console" }
```

**Use Token to request MSRS download:**

```
url=https://msrsapp.pjm.com/msrs/rest/secure/download/reports?shortName=TransmissionEnhancementChargeSummary&version=L&format=C&start=08/01/2018&stop=08/25/2018'
```

```
Request.Open "GET", url, False
Request.SetRequestHeader "Cookie", "pjmauth=" & token
Request.SetRequestHeader "Suma-Account", org -- Suma-Account header is used to specify org other than employer org to download for.
Suma-Account header can be removed for non-SUMA accounts.
Request.Send
```

Where token is retrieved Token from SSO  
org = organization shortname

For requesting ALL accounts associated with SUMA user url should include requestAll=true parameter:



## Coding Examples for Browserless HTTP Request with Token Based Authentication

```
url=https://msrsapp.pjm.com/msrs/rest/secure/download/reports?shortName=TransmissionEnhancementChargeSummary&version=L&format=C&start=08/01/2018&stop=08/25/2018'&requestAll=true
```

**Response can be retrieved using preferred method from request object:**

```
Dim ResponseArray as Variant
```

```
ResponseArray = Request.ResponseBody
```

```
Dim ResponseText as String
```

```
ResponseText = Request.ResponseText
```