

Factura Electrónica: Ventajas Biblioteca DLL LibPyAfipWs

La librería LibPyAfipWs es una DLL escrita en C que permite acceder a los webservices de AFIP / ANMAT y en Windows y Linux, compatible con varios lenguajes (Visual Basic, Visual Fox Pro, Delphi, PHP, .Net, Java, C / C++, etc.) Al no ser un OCX ni objeto COM, es muy versátil (igualmente ver OcxFacturaElectronica y PyAfipWs para una alternativa más dinámica para lenguajes modernos).

La biblioteca puede usarse de varias formas que a continuación se detallan (acceso en tiempo de ejecución, compilación y enlace, declaración en VB y VFP, etc.)

Para más información sobre los métodos soportados, ver LibPyAfipWs. Para más información, ver ManualPyAfipWs, FacturaElectronica y PyAfipWs

Acceso en tiempo de ejecución

La librería se puede acceder desde C o lenguajes similares simplemente llamando a la función LoadLibrary("ruta.dll") y luego se accede a las funciones con GetProcAddress. Es importante notar que este método no necesita que se linkee a la librería en tiempo de ejecución, por lo que es muy flexible para desarrollo.

Ejemplo en lenguaje C estándar para crear un ticket de acceso:

```
#include <windows.h>

int main(int argc, char *argv[]) {
    BSTR tra, cms, ta, ret;
    void *wsfev1;
    bool ok;
    long nro;
    HINSTANCE hPyAfipWsDll;
    FARPROC lpFunc, lpFree;

    /* cargo la librería y obtengo la referencia (poner ruta completa) */
    hPyAfipWsDll = LoadLibrary("../LIBPYAFIPWS.DLL");
    if (hPyAfipWsDll != NULL) {
        /* obtengo los punteros a las funciones exportadas en la librería */
        lpFunc = GetProcAddress(hPyAfipWsDll, "WSAA_CreateTRA");
        lpFree = GetProcAddress(hPyAfipWsDll, "PYAFIPWS_Free");
        if (lpFunc != (FARPROC) NULL) {
            /* llamo al método de la DLL para crear el ticket de req. de acceso */
            tra = (*lpFunc)("wsfe", (long)3600);
            printf("TRA: %s\n", tra);
            /* libero la memoria alojada por el string devuelto */
            (*lpFree)(tra);
        }
    }
    FreeLibrary(hPyAfipWsDll);
}
```

Ver libpyafipws_dll_test.c (ejemplo completo)

Compilación y enlace

La librería se puede usar utilizando el encabezado `libpyafipws.h` que define las funciones, `libpyafipws.lib` para linkearlo. Ejemplo en lenguaje C estándar:

```
#include "libpyafipws.h"

int main(int argc, char *argv[]) {
    BSTR tra, cms, ta;
    void *wsfev1;

    /* Generar ticket de requerimiento de acceso */
    tra = WSAA_CreateTRA("wsfe", 999);
    printf("TRA:\n%s\n", tra);
    /* Firmar criptograficamente el mensaje */
    cms = WSAA_SignTRA((char*) tra, "reingart.crt", "reingart.key");
    printf("CMS:\n%s\n", cms);
    /* Llamar al webservice y obtener el ticket de acceso */
    ta = WSAA_LoginCMS((char*) cms);
    printf("TA:\n%s\n", ta);
}
```

Ver [factura_electronica.c](#) (ejemplo completo)

C# (.Net) con InteropServices

Desde .NET se puede llamar a las funciones de la librería usando los servicios de interoperabilidad (DllImport en `System.Runtime.InteropServices`):

```
using System;
using System.Text;
using System.Runtime.InteropServices;

namespace ConsoleApplication1
{
    class Program
    {
        [DllImport("F:\\\\LIBPYAFIPWS.DLL")]
        private static extern string WSAA_CreateTRA(
            string service,
            long ttl
        );
        static void Main(string[] args)
        {
            string tra;
            tra = WSAA_CreateTRA("wsfe", 3600);
            Console.WriteLine("TRA = {0}", tra);
            Console.ReadLine();
        }
    }
}
```

Ver [factura_electronica.cs](#) (ejemplo completo)

Nota: Para VB.NET se recomienda la interfaz por objetos (Component Object Model): [PyAfipWs](#), ver [wsfev1.vb](#) (ejemplo completo)

Visual Basic, Visual Fox Pro y similares

En lenguajes modernos, se puede declarar la función externa, por ej. en VB (cambiar . . \ por la ruta completa, donde este la DLL y el resto de los archivos de instalación):

```
Function WSAA_CreateTRA Lib "..\libpyafipws.dll" (ByVal service As String, ByVal ttl As Long) As String
Function WSAA_SignTRA Lib "..\libpyafipws.dll" (ByVal tra As String, ByVal cert As String, ByVal pk As String)
Function WSAA_LoginCMS Lib "..\libpyafipws.dll" (ByVal tra As String) As String
```

```
Generar un Ticket de Requerimiento de Acceso (TRA) para WSFEv1
ttl = 36000 ' tiempo de vida = 10hs hasta expiración
tra = WSAA_CreateTRA("wsfe", ttl)
```

```
Especificar la ubicacion de los archivos certificado y clave privada
Path = CurDir() + "\"
Certificado: certificado es el firmado por la AFIP
ClavePrivada: la clave privada usada para crear el certificado
Certificado = "..\reingart.crt" ' certificado de prueba
ClavePrivada = "..\reingart.key" ' clave privada de prueba
```

```
Generar el mensaje firmado (CMS)
cms = WSAA_SignTRA(tra, Path + Certificado, Path + ClavePrivada)
MsgBox cms
```

```
Enviar al webservice de autenticación:
response = WSAA_LoginCMS(cms)
```

```
Extrair el ticket de acceso, ToKen y Sign de autorización
MsgBox response
```

Ver [libpyafipws_dll_test.bas](#) (ejemplo completo)

Para Visual Fox Pro sería similar, declarando la función exportada por la DLL de la siguiente manera:

```
DECLARE STRING WSAA_CreateTRA IN ..\LIBPYAFIPWS STRING @ service, LONG @ ttl

&& Generar un Ticket de Requerimiento de Acceso (TRA) para WSFEv1
ttl = 36000 && tiempo de vida = 10hs hasta expiración
tra = WSAA_CreateTRA("wsfe", ttl)

? tra
```

Ver [libpyafipws_dll_test.prg](#) (ejemplo completo)

Nota: Para VB y VFP se recomienda la interfaz por objetos (Component Object Model): [PyAfipWs](#)

Contacto

Para mayor información, consultar por mail a facturaelectronica@sistemasagiles.com.ar o telefónicamente al 15-3048-9211

Para soporte de la comunidad, revisar la [lista de temas](#) y/o [crear uno nuevo](#) Para novedades y consultas generales, puede usar el [Foro Público](#)

PyAfpWs Copyright 2008, 2009, 2010, 2012, 2013 por MarianoReingart