

# llaveCertificado

## Requerimientos

### Windows (W7)

- **Openssl**  
<http://www.slproweb.com/products/Win32OpenSSL.html>  
[Win32 OpenSSL v1.0.0g Light](#)  
[Visual C++ 2008 Redistributables](#)
- **opensslKey.cs**  
<http://www.jensign.com/opensslkey/opensslkey.cs>
- **Visual Studio 2010**

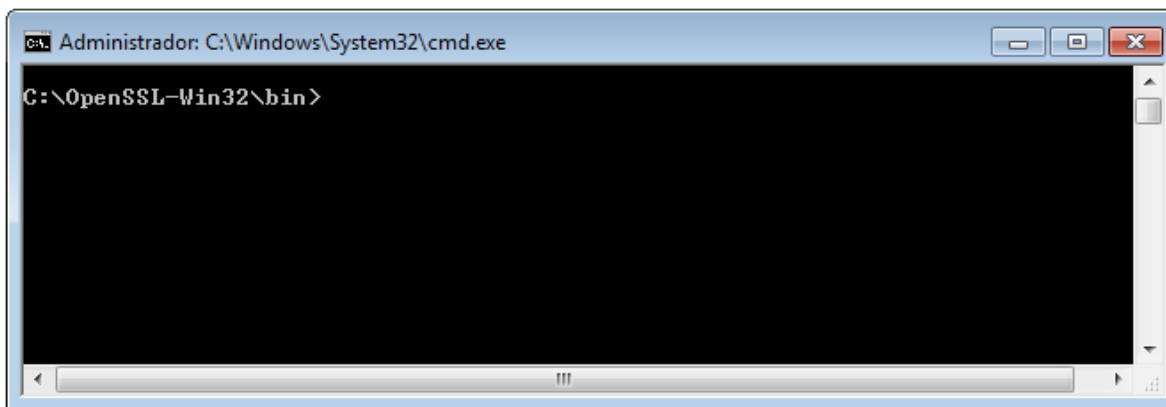
## Obtención llaveCertificado Windows

### Openssl

Instalamos Visual C++ 2008 Redistributables, seguidamente Win32 OpenSSL v1.0.0g Light

Despues de la instalació:

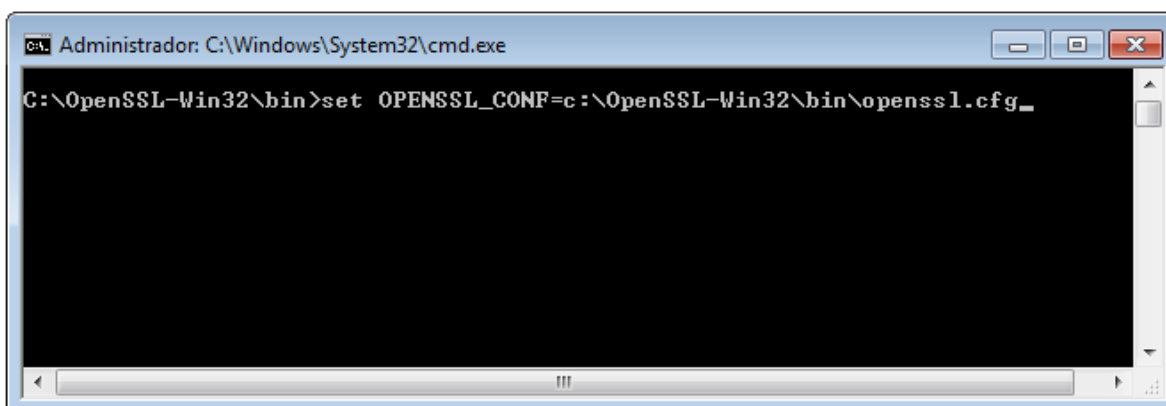
Ejecutamos **command** como administrador,y nos situamos en la carpeta:



```
Administrador: C:\Windows\System32\cmd.exe
C:\OpenSSL-Win32\bin>
```

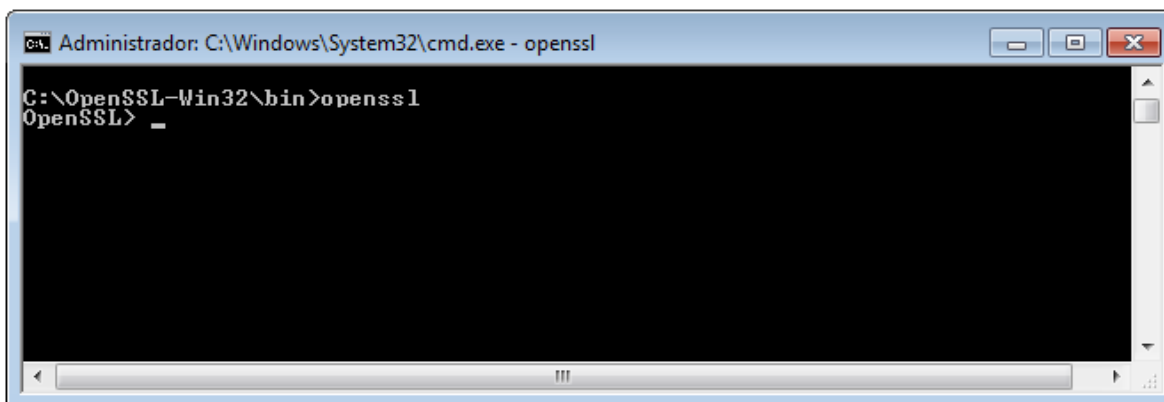
**NOTA:** Si aparece un mensaje de que no se puede acceder al archivo de configuración(openssl.cfg) , establecemos lo siguiente:

```
set OPENSSL_CONF=c:\[Directorio donde se instalo openssl]\bin\openssl.cfg
```



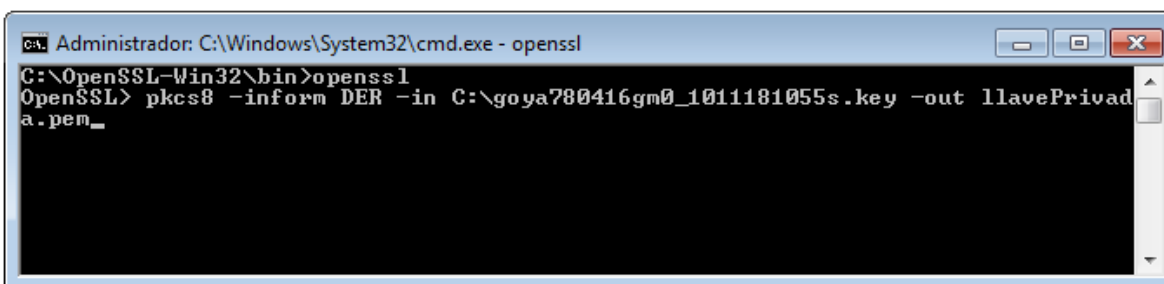
```
Administrador: C:\Windows\System32\cmd.exe
C:\OpenSSL-Win32\bin>set OPENSSL_CONF=c:\OpenSSL-Win32\bin\openssl.cfg_
```

Ejecutamos OpenSSL



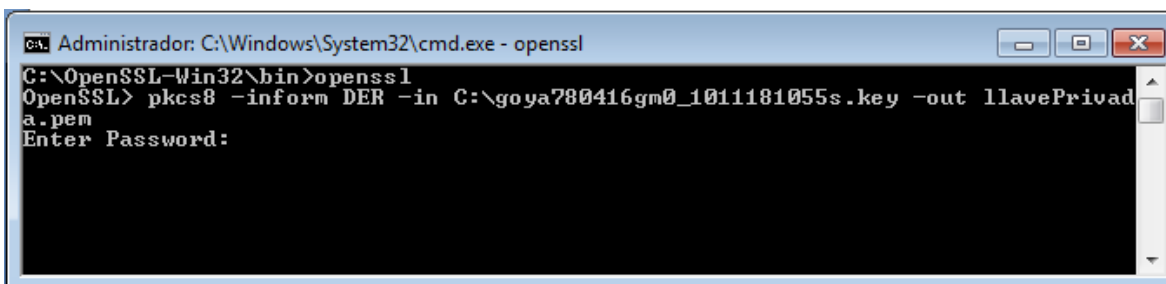
```
Administrador: C:\Windows\System32\cmd.exe - openssl
C:\OpenSSL-Win32\bin>openssl
OpenSSL> _
```

Convertimos llaveprivada a .pem



```
Administrador: C:\Windows\System32\cmd.exe - openssl
C:\OpenSSL-Win32\bin>openssl
OpenSSL> pkcs8 -inform DER -in C:\goya780416gm0_1011181055s.key -out llavePrivada.pem
OpenSSL> _
```

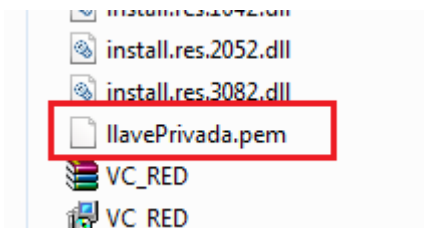
Establecemos contraseña: **12345678a**



```
Administrador: C:\Windows\System32\cmd.exe - openssl
C:\OpenSSL-Win32\bin>openssl
OpenSSL> pkcs8 -inform DER -in C:\goya780416gm0_1011181055s.key -out llavePrivada.pem
Enter Password:
```

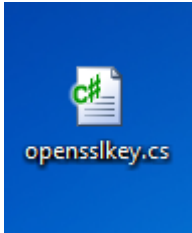
Escribimos **exit** para salir de openssl.

C:\

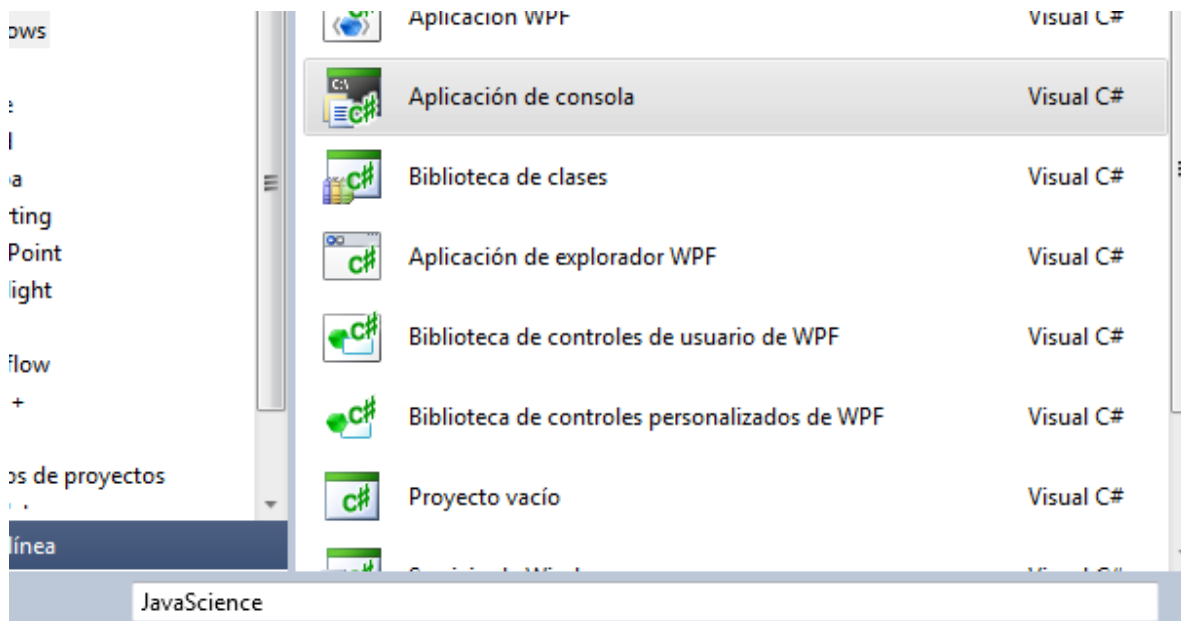


## opensslKey.cs

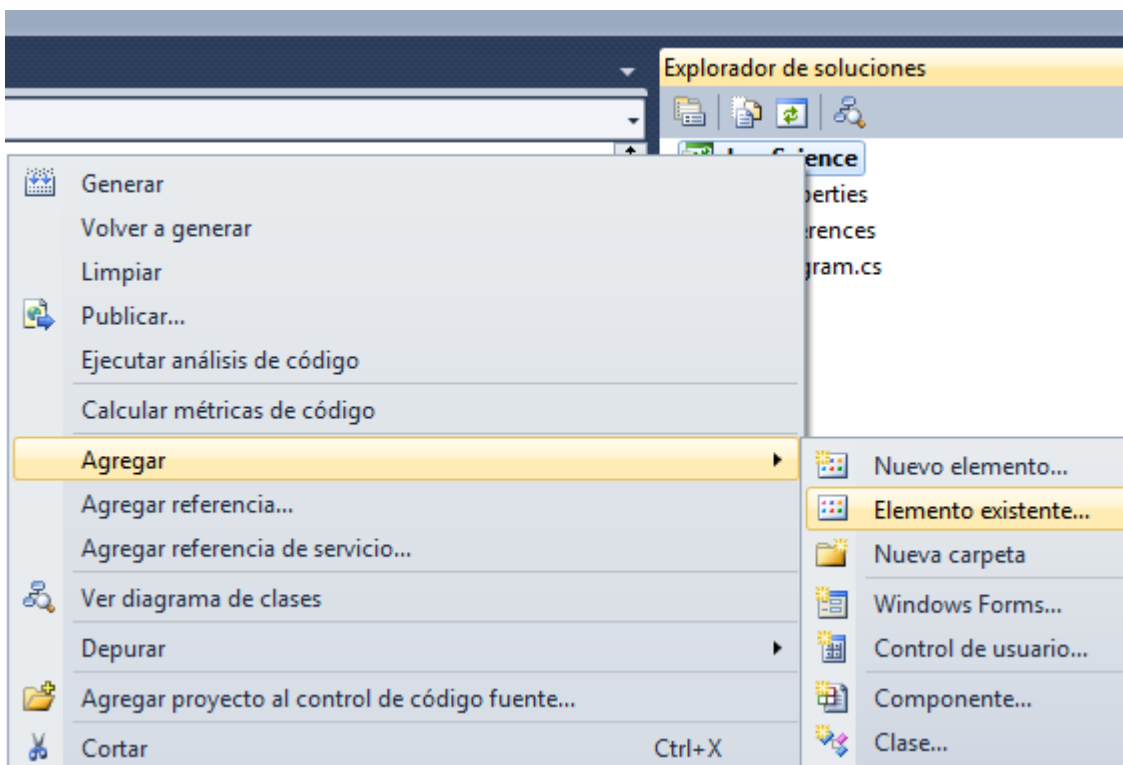
Descargamos opensslKey.cs



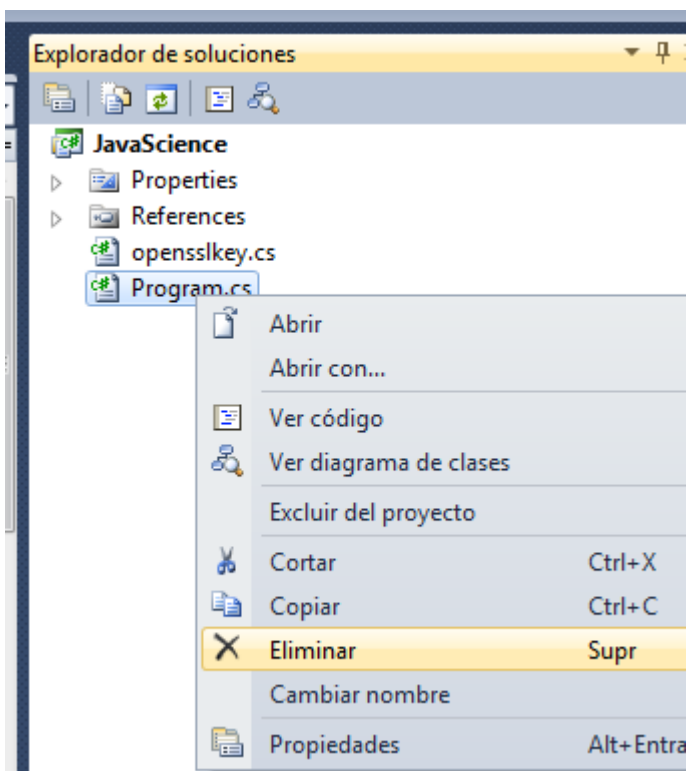
Creamos un nuevo proyecto aplicación de consola, establecemos el nombre JavaScience



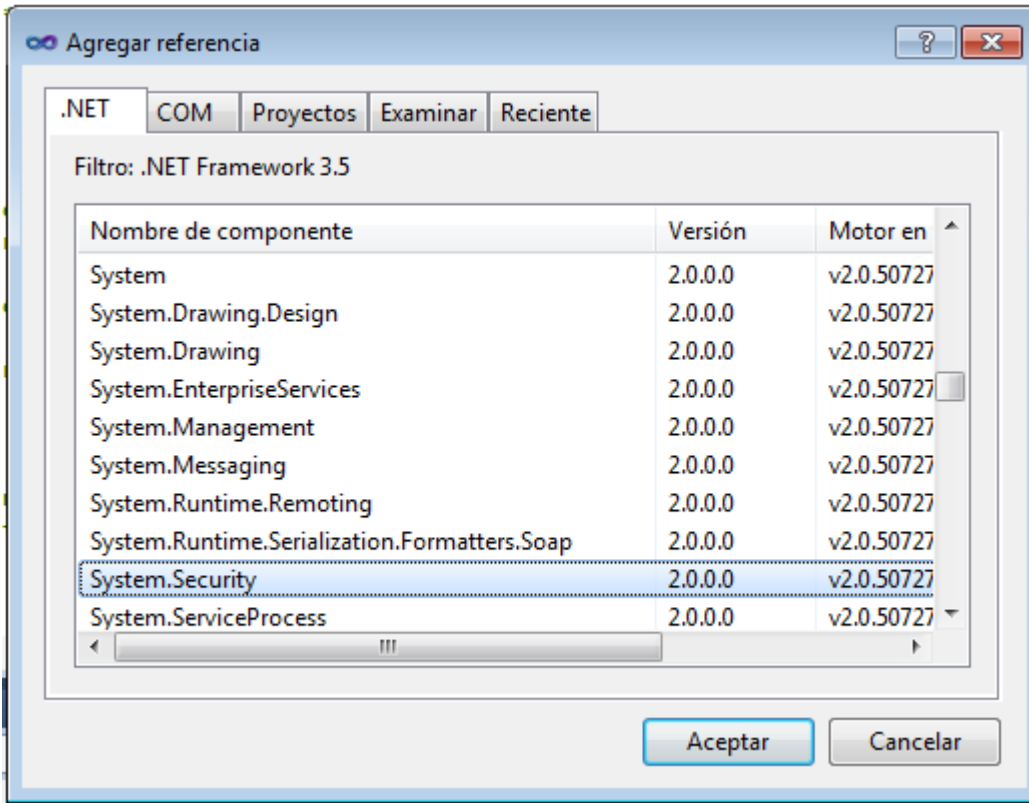
Agregamos openssl.cs



Eliminamos Program.cs



Click derecho sobre proyecto > agregar referencia **System.Security**



Modificaremos esta sección de código:

```
else if(pemstr.StartsWith(pemp8header) && pemstr.EndsWith(pemp8footer))
{
    Console.WriteLine("Trying to decode and parse as PEM PKCS #8 PrivateKeyInfo ..");
    pkcs8privatekey = DecodePkcs8PrivateKey(pemstr);
    if(pkcs8privatekey != null)
    {
        if(verbose)
            showBytes("\nPKCS #8 PrivateKeyInfo", pkcs8privatekey);
        //PutFileBytes("PrivateKeyInfo", pkcs8privatekey, pkcs8privatekey.Length);
        RSACryptoServiceProvider rsa = DecodePrivateKeyInfo(pkcs8privatekey);
        if(rsa !=null)
        {
            Console.WriteLine("\nCreated an RSACryptoServiceProvider instance\n");
            String xmlprivatekey =rsa.ToXmlString(true);
            Console.WriteLine("\nXML RSA private key: {0} bits\n{1}\n", rsa.KeySize, xmlprivatekey);
            ProcessRSA(rsa);
        }
        else
            Console.WriteLine("\nFailed to create an RSACryptoServiceProvider");
    }
}
```

Añadiendo:

```
_llaveprivada = Convert.ToBase64String(Encoding.UTF8.GetBytes(xmlprivatekey.ToString()));
```

```

else if(pemstr.StartsWith(pemp8header) && pemstr.EndsWith(pemp8footer))
{
    Console.WriteLine("Trying to decode and parse as PEM PKCS #8 PrivateKeyInfo ..");
    pkcs8privatekey = DecodePkcs8PrivateKey(pemstr);
    if(pkcs8privatekey != null)
    {
        if(verbose)
            showBytes("\nPKCS #8 PrivateKeyInfo", pkcs8privatekey);
        //PutFileBytes("PrivateKeyInfo", pkcs8privatekey, pkcs8privatekey.Length);
        RSACryptoServiceProvider rsa = DecodePrivateKeyInfo(pkcs8privatekey);
        if(rsa !=null)
        {
            Console.WriteLine("\nCreated an RSACryptoServiceProvider instance\n");
            String xmlprivatekey =rsa.ToXmlString(true);

////////Pasamos XML a Base64

string _llaveprivada = Convert.ToBase64String(Encoding.UTF8.GetBytes(xmlprivatekey.ToString()));

////////////////////////////////////

            Console.WriteLine("\nXML RSA private key: {0} bits\n{1}\n", rsa.KeySize, xmlprivatekey);
            ProcessRSA(rsa);
        }
        else
            Console.WriteLine("\nFailed to create an RSACryptoServiceProvider");
    }
}

```

Para no añadir mas código y obtener en runtime la llave privada, iniciamos una instancia Clic derecho sobre el proyecto > Depurar > crear nueva instancia

Establecemos un Breakpoint

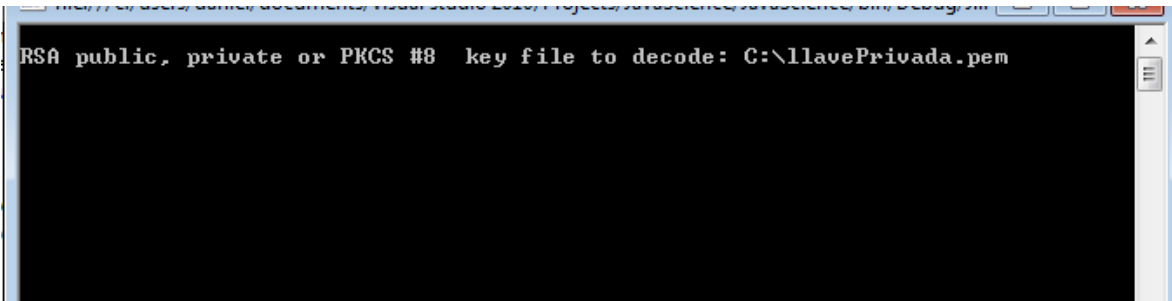


```

{
    Console.WriteLine("\nCreated an RSACryptoServiceProvider instance\n");
    String xmlprivatekey =rsa.ToXmlString(true);
    string _llaveprivada = Convert.ToBase64String(Encoding.UTF8.GetBytes
    Console.WriteLine("\nXML RSA private key: {0} bits\n{1}\n", rsa.Key
    ProcessRSA(rsa);
}

```

Establecemos Ruta de archivo .pem



```
{  
    Console.WriteLine("\nCreated an RSACryptoServiceProvider instance\n");  
    String xmlprivatekey = rsa.ToXmlString(true);  
    string llaveprivada = Convert.ToBase64String(Encoding.UTF8.GetBytes(xmlprivatekey));  
    Console.WriteLine("\nllaveprivada: " + llaveprivada);  
    ProcessRSA(rsa);  
}
```

Click en la lupa

