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, seguimdamente Win32 OpenSSL v1.0.0g Light

Despues de la instalació:

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



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



Ejecutamos OpenSSL

Administrador: C:\Windows\System32\cmd.exe - openssl	
C:\OpenSSL-Win32\bin>openss1 OpenSSL> _	Â
III	• • a

Convertimos llaveprivada a .pem



Establecemos contraseña: 12345678a



Escribirmos exit para salir de openssl.

C:\



opensslKey.cs

Descargamos opensslKey.cs



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

ows		Aplication WPF	Visual C#	
:		Aplicación de consola	Visual C#	
a E	c ♯	Biblioteca de clases	Visual C#	
Point	[∞] c#	Aplicación de explorador WPF	Visual C#	
low	~ C#	Biblioteca de controles de usuario de WPF	Visual C#	
+	€ C#	Biblioteca de controles personalizados de WPF	Visual C#	
os de proyectos	C#	Proyecto vacío	Visual C#	
línea		~ ··· · · ···· ·		-
JavaScience				

Agregamos openssl.cs

			_		
	 Explorador de soluciones 				
	- 🕞 🗟 🖓 🕢				
		+	ence		
	Generar		ertie	'S	
	Volver a generar		renc	es	
	Limpiar		Jram	.CS	
4	Publicar				
	Ejecutar análisis de código				
	Calcular métricas de código				
	Agregar	•	-	Nuevo elemento	
	Agregar referencia			Elemento existente	
	Agregar referencia de servicio		Ľ	Nueva carpeta	
æ,	Ver diagrama de clases		1	Windows Forms	
	Depurar	•	-	Control de usuario	
1	Agregar proyecto al control de código fuente		-	Componente	
¥	Cortar	Ctrl+X	- 23	Clase	

Eliminamos Program.cs



NET COM Proyectos Examinar Reciente		
Filtro: .NET Framework 3.5		
Nombre de componente	Versión	Motor en 🔺
System	2.0.0.0	v2.0.50727
System.Drawing.Design	2.0.0.0	v2.0.50727
System.Drawing	2.0.0.0	v2.0.50727
System.EnterpriseServices	2.0.0.0	v2.0.50727
System.Management	2.0.0.0	v2.0.50727
System.Messaging	2.0.0.0	v2.0.50727
System.Runtime.Remoting	2.0.0.0	v2.0.50727
System.Runtime.Serialization.Formatters.Soap	2.0.0.0	v2.0.50727
System.Security	2.0.0.0	v2.0.50727
System.ServiceProcess	2.0.0.0	v2.0.50727 -
•		4

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()));

////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
 ProcessPSA(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(xmlpr
 Consol Ilaveprivada TPFJTQUtleVZhbHVIPjxNb2R1bHVzPjFLSGpadHFrRXYxRUNUZ3ViUGt
 ProcessRSA(rsa);
}

Click en la lupa

Visualizador de texto	x
Expresión: _llaveprivada	
Valor: PFJTQUtleVZhbHVlPjxNb2R1bHVzPjFLSGpadHFrRXYxRUN UZ3ViUGtCejZLckYyb1Z5Y0t5anREQ0hHT3NyMGlTYzhwU1	-
JWVThMMEJGSXhNWDVYM2JjMDBDNkJ4clpEby9mL0NJanVzb m1TWDdkTzdTWE0wRUM2VkJlQmg0UGNIejljVlRROXRMSk1u dXhINjB4a3lRWFFDbW9ZUFV4V050TFg1TjhOV213UlVHbGZ vOVNGTDhWZ3NQM1g4Ym5aaz08L01vZHVsdXM +PEV4cG9uZW50PkFRQUI8L0V4cG9uZW50PjxQPjlnM010Rl NWVkU5aVlPeWlUV3VYOU9xZVRmUERHY203WFZhRX1FNWR2V	E
ms2bFAzWHZUbzR3N3ZONUNwYnJENGZwZzRCdzN6WjZZdTli bTY4K0hJb2J3PT08L1A+PFE +M1RvODdNT3JLTDdSdktQSTNJeTFqMWM0YVNET1k3R1FPcy 9pWkpXOGVSckUwcUVGOWphNk1xWTV6aE1SWG5kY21hM0ZjU EZ4ZnFmN0ZjeldSRmRPZHc9PTwvUT48RFA +alhPcGR4SVpOdnJjSjdSaUtmZnN40VFKWmU3NC9sNUc1SV bVkUEMV(NNPDicrO01tWkkhU151UTM21ktidkEES2aUL0110	
NVBUFMVINKRISPQØJtwixhuleluTMSUytjakF5S3pULØILU F11MEhDUldtYzNJKzBjN1E9PTwvRFA Image: Constraint of the state of t	•