Autenticación SII - Firma Electrónica

Aprende a autenticarte en el SII usando tu firma electrónica a través de Swagger

Anonymous · 27/08/2026

Autenticación SII - Firma Electrónica

La firma electrónica es el método más seguro para autenticarse en el SII. A diferencia del método RUT/Clave, la firma electrónica:

- No expone tu contraseña del SII
- Es obligatoria para ciertos trámites
- Permite automatización más segura

Métodos de Autenticación con Firma

API Gateway ofrece **dos formas** de enviar tu firma electrónica:

Método 1: cert-data y pkey-data (Recomendado)

Envías por separado:

- **cert-data**: Certificado público en formato PEM
- **pkey-data**: Llave privada en formato PEM

Método 2: file-data y file-pass

Envías el archivo completo:

- **file-data**: Archivo .p12/.pfx codificado en Base64
- **file-pass**: Contraseña del archivo

Preparación del Certificado

Convertir .p12 a PEM

Si tienes un archivo `.p12` o `.pfx`, necesitas convertirlo a formato PEM:

Opción A: Usar API Gateway (Más fácil)

1. Ingresa a tu perfil en API Gateway
2. Busca la sección “Utilidades”
3. Sube tu archivo .p12/.pfx
4. Ingresa la contraseña
5. Obtendrás el cert-data y pkey-data listos para usar

Opción B: Usar OpenSSL

```
openssl pkcs12 -in firma.p12 -out firma.crt -nodes
```

Este comando generará un archivo con:

```
-----BEGIN CERTIFICATE-----
[Tu certificado aquí]
-----END CERTIFICATE-----
-----BEGIN PRIVATE KEY-----
[Tu llave privada aquí]
-----END PRIVATE KEY-----
```

Usando Swagger con Firma Electrónica

Método 1: Con cert-data y pkey-data

1. **Localiza un endpoint** que requiera autenticación SII
2. **Haz clic en “Try it out”**
3. **En el body, estructura el JSON**:

```
{
  "auth": {
    "cert": {
      "cert-data": "-----BEGIN CERTIFICATE-----\nMIIG...[tu certificado]...XYZ\n-----
END CERTIFICATE-----",
      "pkey-data": "-----BEGIN PRIVATE KEY-----\nMIIE...[tu llave privada]...ABC\n----
-END PRIVATE KEY-----"
    }
  }
}
```

4. **Ejecuta** la petición

Método 2: Con file-data y file-pass

1. **Codifica tu archivo .p12 en Base64**:
    - Puedes usar herramientas online o comandos del sistema
    - En Linux/Mac: `base64 -i firma.p12 -o firma_base64.txt`
2. **Estructura el JSON**:

```
{
  "auth": {
    "cert": {
      "file-data": "[contenido base64 del archivo]",
      "file-pass": "contraseña_del_p12"
    }
  }
}
```

Comparación de Métodos

Aspecto	cert-data/pkey-data	file-data/file-pass
Seguridad	Más seguro (sin contraseña)	Incluye contraseña
Preparación	Requiere conversión	Directo con .p12
Rendimiento	Más rápido	Conversión en cada request
Recomendado para	Producción	Pruebas rápidas

Verificando en Swagger

Respuesta Exitosa

Si la autenticación es correcta, verás:

- **Status**: 200 OK
- **Body**: Datos solicitados del SII

Errores Comunes

- Certificado mal formateado

- Firma vencida
- Estructura incorrecta

Tips Importantes

1. Formato de los Saltos de Línea

 Incorrecto: Copiar/pegar sin saltos

```
"cert-data": "-----BEGIN CERTIFICATE-----MIIG...XYZ-----END CERTIFICATE-----"
```

 Correcto: Con \n en cada línea

```
"cert-data": "-----BEGIN CERTIFICATE-----\nMIIG...\n...XYZ\n-----END CERTIFICATE-----"
```

2. Validar el Certificado

Antes de usar en producción:

- Verifica que no esté expirado
- Confirma que sea el certificado correcto
- Prueba primero en un endpoint simple

3. Seguridad

- **Nunca** guardes las credenciales en el código
- **Usa** variables de entorno en producción

Last updated on 21/08/2026