# Autenticación SII - Firma Electrónica

Aprende a autenticarte en el SII usando tu firma electrónica desde la documentación interactiva

Anonymous · 27/08/2026

# Autenticación SII - Firma Electrónica

Con la **firma electrónica** (certificado digital) te autenticas en el SII **sin** enviar tu clave tributaria en el mismo flujo que con RUT/clave. Es útil si quieres evitar manejar la contraseña del SII en integraciones, si el trámite o consulta **exige** certificado según las reglas del SII, o si buscas un esquema más alineado con buenas prácticas de seguridad.

Como en Navegando Swagger UI, llevas **`Authorization: Token …`** en cabecera (token de conexión) y en el **cuerpo** el bloque **`auth`** con el certificado (firma).

## Token de conexión (siempre)

Toda petición incluye:

```
Authorization: Token <tu_token>
```

## Dos formas de enviar el certificado en el JSON

API Gateway admite **dos variantes** dentro de `auth.cert`:

**Método 1: `cert-data` y `pkey-data` (recomendado)**

- **cert-data** es el certificado público en **PEM** (bloque `BEGIN CERTIFICATE`);
- **pkey-data** es la llave privada en **PEM** (bloque `BEGIN PRIVATE KEY`).

**Método 2: `file-data` y `file-pass`**

- **file-data** es el archivo `.p12` / `.pfx` codificado en **Base64**;
- **file-pass** es la contraseña de ese archivo.

## Preparación del certificado

Para autenticar con el SII usando firma electrónica, puedes utilizar las siguientes opciones:

### Opción A: Herramientas en la aplicación

1. Entra a **app.apigateway.cl**.
2. **Gestionar conexión** → pestaña **Certificado**.
3. Allí puedes obtener o convertir material para usar en **PEM** o seguir las instrucciones en pantalla.

Enlace directo al formulario de formato de certificado: Gestionar conexión

### Opción B: OpenSSL en tu computador

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

Luego **separas** en un editor de texto los bloques:

```
-----BEGIN CERTIFICATE-----
...
-----END CERTIFICATE-----
-----BEGIN PRIVATE KEY-----
...
-----END PRIVATE KEY-----
```

## Uso en la documentación interactiva

### Método 1: PEM en el body

1. Localiza un endpoint que requiera autenticación SII.
2. **Authorize** con tu **token de conexión**.
3. **Try it out** y pega un JSON como:

```
{
  "auth": {
    "cert": {
      "cert-data": "-----BEGIN CERTIFICATE-----\nMIIG...\n-----END CERTIFICATE-----",
      "pkey-data": "-----BEGIN PRIVATE KEY-----\nMIIE...\n-----END PRIVATE KEY-----"
    }
  }
}
```

4. **Execute**

### Método 2: `.p12` en Base64

1. Genera Base64 del archivo (ejemplo en Mac/Linux):

```
base64 -i firma.p12 | tr -d '\n'
```

2. Body:

```
{
  "auth": {
    "cert": {
      "file-data": "[pega aquí el base64 completo]",
      "file-pass": "contraseña_del_p12"
    }
  }
}
```

## Comparación rápida

| Aspecto | cert-data / pkey-data | file-data / file-pass |
|---|---|---|
| **Secretos** | Sueles evitar reenviar la contraseña del `.p12` en cada llamada si ya extrajiste PEM | Incluye `file-pass` en la petición |
| **Preparación** | Extraer PEM | Más directo si solo tienes `.p12` |
| **Uso típico** | Integraciones estables | Pruebas rápidas |

## Respuesta esperada y errores frecuentes

#### Si la autenticación es correcta

Suele responder con código **2xx** (a menudo **200**) y un cuerpo con los datos solicitados al SII.

#### Problemas habituales

PEM mal pegado en JSON (**faltan** los `\n` entre líneas), certificado **vencido** o revocado, o estructura JSON incompleta (falta `auth.cert` o campos requeridos).

## Tips importantes

#### 1. Saltos de línea en JSON

 Todo en una sola línea sin `\n`:

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

 Con `\n` como separador de líneas PEM:

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

### 2. Antes de producción

Confirma la **fecha de vencimiento** del certificado, prueba primero un endpoint sencillo y limita quién puede ver los PEM o `.p12` en tu organización.

### 3. Seguridad

No subas llaves ni `.p12` a repositorios públicos; usa secretos del entorno o un vault en servidores.

## Relación con la sesión y `auth_cache`

Si aparecen errores de sesión o cabeceras como `X-Auth-Session-Problem`, revisa Autenticación SII — RUT y clave (sección **Control de caché de sesión**) y Manejo de errores.

Last updated on 21/08/2026